



A Detailed Analysis of a Systematic Review About Requirements Engineering Processes for Multi-agent Systems

Giovane D'Avila Mendonça^(✉) , Iderli Pereira de Souza Filho ,
and Gilleanes Thorwald Araujo Guedes 

Pampa Federal University, Av. Tiaraju, 810, Alegrete, Brazil
{giovaniemendonca, iderlisouza.aluno,
gilleanesguedes}@unipampa.edu.br

Abstract. Requirements Engineering is a crucial area of Software Engineering, as it is through it that what a system should actually do is established, what are its functionalities and what is the expected behavior for the software. It is also through the requirements definition that the basic criteria to determine the software quality are established. All this is equally true for multiagent systems, systems composed of autonomous agents that can work together to solve complex problems. Nevertheless, requirements engineering for this kind of system presents its own challenges, as this type of system has specific requirements that are not normally found in other types of software. Taking this into account, we first performed a systematic literature review in order to determine the current state of the development processes that support requirements engineering for multiagent systems, classifying their coverage regarding the requirements engineering subareas and their support to the BDI model - a widely used model for the cognitive agents development. However, due to the lack of space, in the original publication we were only able to discuss the retrieved studies in a general way. Therefore, in this new work, we discuss the studies of the first review in more depth, emphasizing mainly the methodologies that support the BDI model and the gaps found after analysing them. Furthermore, we highlight the possibilities for further research based on the study of the recovered works.

Keywords: Requirements engineering · Multi-Agent Systems · BDI model · Systematic review

1 Introduction

Agents are intelligent entities that act in a flexible and autonomous way in decision making. A multiagent system (MAS) contains several agents acting in a system to solve problems beyond the capability of just one agent [56]. Multiagent systems have been shown to be good alternatives to deal with complex systems, since the complexity can be divided and assigned to several agents specialized in a given facet of the problem [31].

However, developing this kind of system also proved to be complex and generated new challenges for software engineering, which led to the emergence of AOSE -

Agent-Oriented Software Engineering, an area that mixes features from both software engineering and Artificial Intelligence [36].

One of the several software engineering subareas is requirements engineering, which aims to elicit, analyze, specify, and validate system requirements, in order to ensure the correct understanding about what a system really should do. According [73], requirements engineering is one of the most vital activities in the entire software development lifecycle, as the success of a software project depends much of how well users requirements have been understood and converted into proper functionalities.

Thus, several requirements engineering processes for multiagent systems have been proposed over the years. These processes aim to offer techniques, methods, and models adapted for this type of system, since multiagent systems present their own complexity, challenges, and particular requirements which differentiate them from other kinds of systems.

In SWEBOK [15] - a reference book in the field of software engineering - requirements engineering is divided into four subareas, Requirements Elicitation, Requirements Analysis, Requirements Specification, and Requirements Validation. We emphasize that each one of these subareas plays an essential role to Requirements Engineering.

This way, in order to understand the state-of-art of requirements engineering processes for multiagent systems, we carried out a literature systematic review [59]. This review analysed the processes mainly regarding the coverage of requirements engineering subareas defined in SWEBOK and its support to the BDI model, one of the most recognized approaches to integrate desired cognitive skills in autonomous agents and which facilitates the description of the relation cause-effect needed to an agent to achieve its goals [5]. Furthermore, this model proved to be one of the most suitable for the development of agents with flexible behavior [4].

However, this systematic review only presented the support for the BDI model and the coverage regarding SWEBOK requirements engineering of the analysed processes in a general way, without discussing these works in more depth. Therefore, this current paper aims to extend the analysis performed in the previous review, describing in detail the support of the processes regarding the BDI model and the gaps found in these processes, both in the BDI model support as in the processes coverage regarding the requirements engineering subareas as defined in SWEBOK.

This work is organized as follows. Section 2 contains the background. Section 3 presents the related works. The research method is described in Sect. 4. In Sect. 5, the results are presented and discussed. Threats to the validity are described in Sect. 6. Next, we present some insights that we gained from this research and, finally, the conclusion and future works.

2 Background

2.1 Requirements Engineering

According to [9], the RE goals are: (I) to identify software requirements, (II) to analyse requirements in order to classify them and to derive additional requirements, as well as to solve conflicts among them (III) to document requirements, and (IV) to validate the documented requirements.

In SWEBOK [15] - a reference book in the area - it is stated that the RE process cover four main subareas: (I) Requirements Elicitation; (II) Requirements Analysis; (III) Requirements Specification; and (IV) Requirements Validation.

Requirements elicitation investigates how to extract requirements and which are its origins. Requirements analysis aims to detect and solve conflicts among the requirements and to discover the system boundaries. Requirements specification, by its turn, produce requirements documents that can be systematically reviewed, evaluated, and approved. Finally, requirements validation evaluates requirements documents to ensure that the requirements be understandable, consistent, and complete.

2.2 Belief-Desire-Intention Model

The Belief-Desire-Intention (BDI) model is a software model developed to programming intelligent agents. It includes beliefs, desires, and intentions in the agent architecture [17], and that, according to [75] has been widely used in MAS development. Moreover, in the last 30 years this model has been the basis for much research on autonomous agents architectures [28].

In this model, Beliefs represent the information state the agent owns, i.e., what he believes to be true about the environment, about itself, and about other agents. Desires represent the agents motivational state. They represent the goals or situations the agent would like to achieve. Finally, the intentions represent desires the agent believes he can achieve and take actions to achieve them [67].

This model allows to the agents a more complex behavior than the reactive models, without the computational overload of the cognitive architectures. Moreover, it is easier to specify knowledge by means of this model [49].

According to [41], concepts of belief and goal perform a central role in the conception and implementation of autonomous agents. The concept of BDI, consider mental attitudes to be fundamental to the agents, where the beliefs are adapted to the environment truths, while in the intentions, the agents try to make the environment to correspond to its goals.

3 Related Works

We discovered some studies that aimed to identify and to evaluate methodologies/processes in the AOSE area. However, these studies do not follow a systematic vision, they are informal literature reviews with subjective comparison criteria.

The study of [40] discusses the state of AOSE methodologies and how to turn them into acceptable products for the industry. This study also presents a methodology classification, dividing them in (I) independent of goal-oriented methodologies and (II) extensions of goal-oriented methodologies to give support to the agent concepts. The study of [76] evaluates agent-oriented software methodologies. The work proposes a comparison frame with four selection groups: concepts, notations, process, and pragmatics. This proposal was evaluated comparing the methodology adequation and its development capacity. For this comparison, were used three methodologies, MaSE (an old version of O-MaSE), Tropos, and Prometheus. Finally, the work of [22] investigates the AOSE methodologies coverage regarding software engineering concepts. However,

besides this work not following a systematic vision, it does not present several methodologies and does not have a wide coverage of requirements engineering.

Regarding systematic reviews, we found the study of [11] that developed a review about requirements engineering in multi-agent systems development. Nevertheless, this study tried to verify which modeling techniques were applied in the requirements engineering for MAS. On the other hand, our work has as its goal to identify the coverage of the requirements engineering process regarding the SWEBOK stages and its adequation to the BDI model.

The Table 1, presents a comparison of the studies related to our work, having as a comparison point the main features of this work. As can be noticed our work covers some characteristics that are different from the previous ones.

Table 1. Comparison of studies related to this work.

Study	Multi-Agent Systems	Focus on Methodologies	Systematic review	Requirements Engineering	BDI Model
[40]	✓	✓			
[76]	✓	✓			
[22]	✓	✓			
[11]	✓		✓	✓	
This Work	✓	✓	✓	✓	✓

4 Research Method

A systematic literature review (SLR) is a research technique whose purpose is identifying, selecting, evaluating, interpreting, and summarizing the available studies considered relevant to the research theme or phenomenon of interest [47]. This technique searches for primary studies related to the theme and provides a deeper synthesis about the data obtained from these studies [48].

A SLR has as its basis a protocol previously defined, that formalizes its execution, beginning by the stipulation of the research questions, passing by establishing the studies inclusion and exclusion criteria, selecting the digital basis for the extraction of works related with the keywords applied during the search in these basis, and concluding with the definition of how the results will be presented [10].

Our review has as its goal to establish the state-of-art of the process/methodologies for MAS development that support in somehow requirements engineering for this kind of system. Our main interest is about how these processes identify and specify the BDI model features in the requirements engineering phase.

4.1 The Research Questions

We defined four research questions for this review. The first research question (RQ1) aims to identify which methodologies/processes support RE for MAS.

The second research question (RQ2) was defined to identify the coverage of the RE by these methodologies. We believe that with this question we can discover possible gaps in the area and that this will allow for future research.

The third research question (RQ3) aims to verify which methodologies support the BDI model. As we stated before, this is a consolidated model in the MAS development and we believe it aggregates better reliability in using the methodologies that support it.

Finally, the fourth question (RQ4) has as its goal to show a wider view of the area needs and to focus on the points that can be approached in future works.

The four research questions are listed below:

- **RQ1:** Which methodologies for the MAS development support a specific requirements engineering (RE) life cycle to this kind of system?
- **RQ2:** Which is the coverage of the requirements engineering by these methodologies taking as a basis the subareas defined by SWEBOK [15]?
- **RQ3:** Which of these methodologies focus on the BDI model during the requirements engineering?
- **RQ4:** Which are the existing gaps in the methodologies that support RE for MAS?

4.2 Identifying and Selecting Primary Studies

To recover relevant works for this study, we built a String containing a set of keywords based on the research questions. This String was adapted to the particularities of each bibliographic basis.

To perform this review, we used bibliographic bases which (I) have a search mechanism based on web; (II) have a mechanism able to use keywords; (III) contain documents from the computer science area; and (IV) their data bases are updated regularly. It is important to highlight that we do not limited the period in which the studies were published

In addition, we have included a book [26] about methodologies for MAS, as well as other classical and known studies. These studies were manually selected by a specialist in the area because we considered that they would not be selected in the search String as they do not present in its title, abstract, and keywords topics related to the requirements engineering, since they are not processes focused on RE, though their life cycles encompass the RE area.

In Table 2 we show the generic String used in the basis. In addition to the search String, we used manual filters in the bibliographic bases. We considered necessary to apply these manual filters because, in some bases, the results obtained were high and many of the studies returned were outside the scope.

For ACM library it was used the filter “Title/Abstract/keywords”; for Engineering Village, “Subject/Title/Abstract”; for IEEE Xplore, All metadata, filters suggested by the base software “agents and multi-agent systems”; for Science Direct, “Subject/Title/Abstract” and “Title/Abstract/keywords” and commands “multiagent OR multi-agent OR agent-based”; for Scopus, “Title/Abstract/keywords”; and for Springer Link it were applied the filters “Filter of the area: Computer science”, “Filter of the subarea: Software Engineering and Artificial intelligence”.

4.3 Inclusion and Exclusion Criteria

The selection criteria have as its objective to identify the primary studies that provide contents to answer the research questions. Thus, firstly the studies were analysed with

Table 2. String generic [58].

String	Conector
("multiagent" OR "multi-agent" OR "multi agent" OR "agent-based" OR "agent society")	AND
("methodology" OR "method" OR "process") ("requirements engineering" OR "requirements elicitation" OR "requirements modeling" OR "requirements analysis" OR "requirements specification")	AND

basis on the title, abstract, and keywords. If there were still doubts about the final classification of a study in relation to the inclusion or exclusion criteria, a specialist would be consulted. These criteria are described in the Table 3.

Table 3. Inclusion and Exclusion Criteria [58].

Criterion	ID	Description
Inclusion	IC1	Does the study presents a methodology or an extension of a methodology for multi-agent systems that contemplates at least one of the requirements engineering subareas defined in the SWEBOK?
Exclusion	EC1	Studies that cover a methodology already included in more recent work
	EC2	Studies that are not a paper or a chapter of book
	EC3	Studies with less than 6 pages
	EC4	Studies that boils down to a case study or methodology evaluation
	EC5	Studies that boils down to a comparison of methodologies
	EC6	Studies that do not present a methodology (or extension of a methodology) for multi-agent systems that contemplate at least one of requirements engineering subareas from SWEBOK
	EC7	Studies that concentrate in other areas of Software Engineering
	EC8	Studies that boils down to the development of a system
	EC9	Studies that present a methodology or extension of a methodology created only to a kind of specific application

4.4 Studies Quality Assessment

We defined two quality criteria to evaluate the relevance of the studies to the scope of this research. These criteria were not used to the exclusion of studies, only for the ranking of studies more relevant. Next we described the two qualitative criteria and the score attributed for each criterion defined.

1. **QC1:** The work supports the BDI model?
Yes (Y): the work fully supports the BDI model;
Partly (P): the work supports at least one of the features of the BDI model;
Not (N): the work does not support the BDI model.
2. **QC2:** the work applies some empirical study (experiment, case study, etc.)?
Yes (Y): the study applies some empirical study;
Not (N): the study does not apply some empirical study.

To establish a quality general index of the selected studies, we attributed scores to each criterion defined, where Yes (Y) corresponds to 1 score, Partly (P) 0.5 score and Not (N) 0 score.

The Table 4, shows the score of each selected study. We noticed that only three studies ([46,60,62]) reached the maximum ranking of 2 scores.

On the other hand, some studies got 0 score ([2,13,35,38]), though these studies did not achieve any score, they were kept because the qualitative criteria were used only for ranking the studies, not for eliminate them.

Table 4. Quality Indexes of the Studies [58].

Study	QC1	QC2	Total	Study	QC1	QC2	Total
[78]	0.50	0.00	0.50	[77]	0.50	1.00	0.00
[1]	0.50	1.00	1.50	[71]	0.50	1.00	1.50
[65]	1.00	0.00	1.00	[39]	0.50	1.00	1.50
[53]	0.00	1.00	1.00	[55]	0.50	1.00	1.50
[6]	0.50	0.00	0.50	[80]	0.50	0.00	0.50
[70]	0.50	1.00	1.50	[81]	0.50	1.00	1.50
[12]	0.50	1.00	1.50	[54]	0.00	1.00	1.00
[44]	0.50	0.00	0.50	[38]	0.00	0.00	0.00
[32]	0.50	1.00	1.50	[7]	0.50	1.00	1.50
[68]	0.50	1.00	1.50	[13]	0.00	0.00	0.00
[42]	0.50	0.00	0.50	[43]	0.00	1.00	1.00
[19]	0.50	0.00	0.50	[33]	0.50	1.00	1.50
[50]	0.50	1.00	1.50	[64]	0.50	1.00	1.50
[66]	0.50	0.00	0.50	[79]	0.50	1.00	1.50
[52]	0.50	0.00	0.50	[69]	0.50	1.00	1.50
[74]	0.50	0.00	0.50	[30]	0.00	1.00	1.00
[2]	0.00	0.00	0.00	[23]	0.50	0.00	0.50
[46]	1.00	1.00	2.00	[27]	0.00	1.00	1.00
[62]	1.00	1.00	2.00	[35]	0.00	0.00	0.00
[57]	1.00	0.00	1.00	[14]	0.00	1.00	1.00
[24]	0.50	0.00	0.50	[29]	0.50	0.00	0.50
[8]	0.50	0.00	0.50	[63]	1.00	0.00	1.00
[60]	1.00	1.00	2.00	[20]	0.50	1.00	1.50
[61]	0.00	1.00	1.00	[21]	0.50	0.00	0.50
[25]	0.50	0.00	0.50	[34]	1.00	0.00	1.00
[18]	0.50	1.00	1.50	[45]	1.00	0.00	1.00
[72]	0.50	1.00	1.50	[51]	0.00	1.00	1.00

4.5 Data Extraction Strategy

When the studies selection process was concluded, the basic information of each paper was registered for data extraction. The extraction was performed using the Google Spreadsheet to capture all the information of each work included, allowing the posterior synthesis.

The data extracted from the included works were analysed in order to answer the research questions. In Sect. 5, these results were exposed and discussed.

4.6 Conducting the Review

The conduction of this systematic review was performed between the months of February and May of 2020. We defined four stages for the studies selection: (I) executing the search String in the bibliographic bases; (II) removing the duplicated studies; (III) applying the inclusion and exclusion criteria to the works; and (IV) reading and extracting the information of the remaining studies of the Stage (III). The studies were read by two reviewers in consultation with a specialist in the area.

In Stage 1, the search String was executed in the bibliographic bases selected for this review. The overview of this stage can be observed in Fig. 1. The conduction began analysing the 1060 works imported from the selected bibliographic bases.

In Stage 2, a total of 247 duplicated studies were removed. In Stage 3, there were applied the inclusion and exclusion criteria based on the reading of the title, abstract, and keywords, resulting in the selection of 53 studies considered promising.

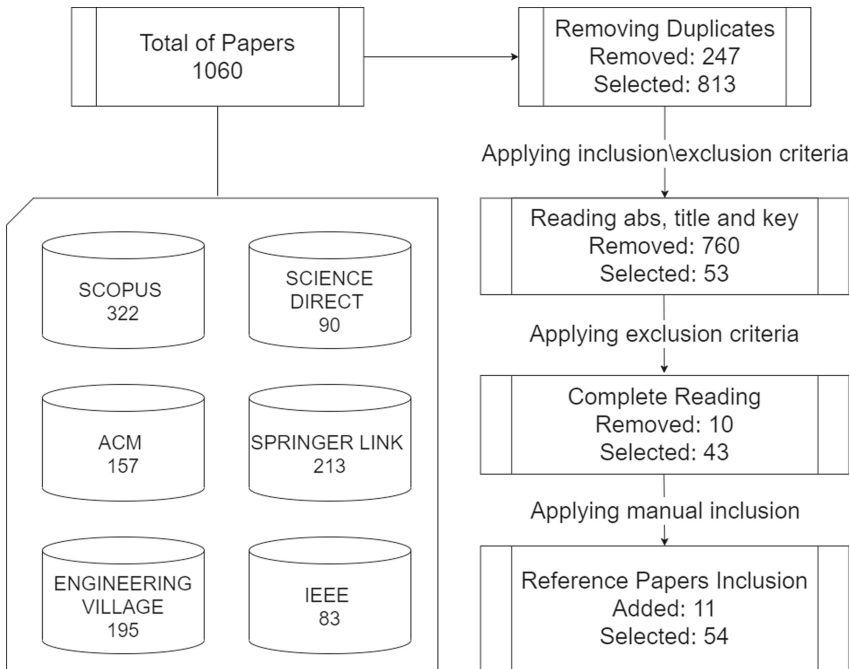


Fig. 1. Search process [58].

To avoid the selection of works that are not fitted in the scope of this review, the 53 studies, selected in the third stage, were completely read, what resulted in the exclusion of 10 works, totalling 43 selected works. The rejected works in this stage were inside of two exclusion criteria:

1. Works that concentrate in other Software Engineering areas: the works excluded that were inside on this criterion were methodologies that worked with MAS only in posterior stages to the requirements engineering. The requirements engineering was performed in a traditional way, not focusing on any particular feature of MAS.
2. Works that cover a methodology already included in a work: for this criterion we selected the most recent work in such a way we can understand the current state of the methodology.

At the end of the conduction Stage, the manually selected studies ([14,20,21,23,27,29,34,35,45,51,63]) were added to the set of papers searched in the bases, according with defined in Sect. 4.2. This resulted in a total of 54 accepted studies.

4.7 Data Extraction

For data extracting in the accepted works, we read them all and tried to identify which SWEBOK RE subareas each work covers, whether the methodology proposed in the study has a well-defined life cycle, whether the RE presented in the study is adequate for MAS, and whether the study supports the BDI model.

The conduction of this stage was performed in pairs, where each researcher read the paper and extracted the information about the issues cited previously. The conflicts between the researchers were decided by a specialist in the area.

5 Results

The relevant information of the selected studies was obtained using the data extraction spreadsheet. The evidence found about each research question are discussed in the next subsections.

5.1 Methodologies Analysed in the Systematic Review

This subsection aims to answer the first research question: “Which methodologies for the MAS development support a specific requirements engineering (RE) life cycle to this kind of system?”

To answer this question, we found 54 methodologies which approached RE for MAS. These studies can be observed in Table 5.

Table 5. Methodologies/Processes that support RE for MAS and its coverage with relation to the SWEBOK subareas [58].

Methodology	Elicitation	Analysis	Specification	Validation
KAOS Extension [78]		✓	✓	
JAAMAS [1]			✓	
Patrizia Ribino [65]			✓	
AGSIRA [53]		✓	✓	
GORMAS [6]		✓	✓	
ATABGE [70]	✓			
RE4Gaia [12]		✓		
Xu Huiying [44]		✓	✓	
REG for AOSE [32]	✓	✓		
Extension GAIA [68]		✓		
B-Tropos [19]	✓	✓	✓	
JONGWON LEE [50]		✓		
Prabhat Ranjan [66]	✓	✓	✓	
SRAMO [52]		✓		
Zhiqi Shen [74]		✓		
SONIA [2]		✓	✓	
BDI ASP [46]		✓		
Tropos [62]	✓	✓	✓	
Cysneiros [57]		✓	✓	✓
Chiung-Hui [24]	✓	✓		
KAOS [8]		✓	✓	
PRACTIONIST [60]		✓	✓	
Murray [61]			✓	
ASPECS [25]		✓	✓	
REF [18]		✓	✓	
Sen and Jain [72]	✓			
Hsieh et al. [43]		✓	✓	
Sutcliffe [77]		✓	✓	
Agile Sen and Jain [71]	✓			
CREWS-EVE [39]	✓		✓	✓
Cao et al. [55]		✓	✓	
HOMER [80]	✓			
Wu et al. [81]	✓			
Liu and Li [54]		✓		
Mahmoud et al. [38]		✓		
Ashamalla et al. [7]	✓	✓		
Consensus [13]		✓	✓	
Hilaire et al. [42]		✓		
Gaur and Soni [33]		✓		
Passos et al. [64]	✓	✓		
PLANT [79]		✓	✓	
Ronald et al. [69]		✓	✓	
aMIAC [30]		✓	✓	
GAIA [23]		✓	✓	
PASSI [27]	✓	✓	✓	
INGENIAS-SCRUM [35]	✓	✓	✓	
ADELFE [14]	✓	✓	✓	✓
O-MaSE [29]		✓	✓	
PROMETHEUS [63]		✓		
MESSAGE [20]		✓	✓	
OSOAD [21]		✓		
COMOMAS [34]		✓		
MAS-COMMONKADS [45]		✓	✓	
MASSIVE [51]	✓	✓	✓	
Total	16	46	31	3

5.2 Coverage in Relation to the Requirements Engineering Subareas Defined in SWEBOK

This subsection aims to answer the second research question: “Which is the coverage of the requirements engineering by these methodologies taking as a basis the subareas defined by SWEBOK [15]?”

From the 54 selected studies we observed that all of them present Requirements Engineering fit for multi-agent systems. Thus, we extracted which RE sub-areas defined in SWEBOK [15] are supported by these studies.

Table 5 shows the 54 studies and the sub-areas that they support. Great part of these studies, 46 in the total, support the sub-area of requirements analysis. While 31 of them support the sub-area of requirements specification.

The sub-area of requirements elicitation, by its turn, is supported by 16 studies.

From these methodologies, Homer methodology (Human Oriented Method for Eliciting Requirements) [80], stood out for being the only one that presented a way to elicit requirements from direct contact with the stakeholders.

Homer operation can be described as the use of organizational metaphors. In these metaphors, during the interview, the Stakeholder must simulate that the system is a company that is hiring new employees to work on a certain problem. This elicitation style aims to more easily discover the agent roles and their goals inside the system. In its execution, Homer presents a series of questions that should be asked to the Stakeholders.

However, Homer methodology presents problems in its application. In addition to not supporting requirements related to the BDI model, there is not a way to easily elicit the agents perceptions and plans. Moreover, this methodology does not provide support for eliciting requirements about external agents, being focused on the system's internal functions.

Another important point is that no methodology contains a way to elicit requirements related to the BDI model. Although Tropos methodology [62] supports the BDI model and has a elicitation phase, Tropos does not detail how its elicitation should be done, differently from Homer.

We also noticed that several of the methodologies do not seek to determine whether the use of multiagent systems is valid, being uncommon a methodology that contains a viability study to establish the need for a multiagent system. Most of the time, methodologies assume that the system under analysis is a multiagent system.

Finally, concerning the sub-area of requirements validation, it is the area that have the lower number of studies, with only 3 of the total supporting validation.

We also noticed that, from these studies, only ADELFE methodology [14] supports the four RE sub-areas (elicitation, analysis, specification, and validation). However, the elicitation in the ADELFE methodology is not suitable for MAS, being applied a traditional elicitation. The features suitable for a MAS began to be presented in the analysis stage. However, this stage does not present the means for validating the documents specific for MAS. ADELFE validates only documents produced by a traditional requirements engineering.

Another important fact that we noticed in the extraction is that only 30 studies presented some empirical experiments for the validation of the methodology.

5.3 Methodologies Supporting the BDI Model

In this subsection we aim to answer the third research question: “Which of these methodologies focus on the BDI model during the requirements engineering?”

We tried to identify which methodologies support the BDI model. We observed that most part of the studies, 43 in the total, support partially the BDI model, i.e., they identify at least one of the features of this model.

These features are: agent beliefs; agent goals/desires; and agent intentions. However, it is necessary to state that the majority of these works do not cite explicitly the BDI model, most of them are goal-oriented methodologies, i.e., they focus on just in one feature of the BDI model and they do not necessarily use this model, but the fact that these studies identify one of the features is useful for our research.

Agents goals were the most identified feature, in most cases in isolation. There are studies that identify intentions, however we noticed that beliefs and intentions are not identified in isolation, they are always accompanied by the identification of their goals.

Another issue to be highlighted is that 11 studies do not present support to the BDI model and only 8 present support for all these features in at least one stage of their requirements engineering. Table 6 presents the methodologies coverage regarding their support to the BDI model.

Next we will present how the 8 methodologies that support totally the BDI model, in some requirements engineering phase, are structured.

The study of Ribino [65], supports the BDI model in the phase related to the requirements analysis. This methodology proposes a modeling of BDI organization for requirements analysis. This analysis is performed through an ontology that aims to represent the problem domain reality and the problem specification incompleteness.

The ontology describes the environment, its main states, and what can be done to achieve/modify these states. This ontology is composed of the Action (specialized in intentional and unintentional action), Concept (specialized in position and object) and Predicate metaclasses. In this ontology, the “predicates” represent the beliefs and states that influence a concept that, in turn, works as a scenario related to the environment.

We understand that this way of representing beliefs weakens a view of BDI agents since, for this kind of agents, beliefs represent the knowledge that an agent or agent role has about the environment, and delimiting the knowledge related only to the environment can result in conflicts in the analysis, since all beliefs are related to the environment and not to a particular agent or agent role.

This methodology allows the identification of goals from patterns discovered by the requirements analysis described textually. There is a set of guidelines that establish how to identify positions, actions, and predicates. However, it is not possible to identify the perceptions associated with a goal or how that goal becomes an intention.

The BDI ASP process [46] appeared to be promissory in its preliminary reading. This process is entirely turned to BDI agents development, besides using as its basis the use-cases production, a technique widely used both in the industry and academia.

Nevertheless, after an analysis of the process, we found errors that could compromise this proposal. The first point is the use of the term intention as a synonym of plan. We believe that this could compromise the support of the BDI model, since an intention is usually related to one or more plans, but intentions and plans are not synonymous. In

Table 6. Coverage of methodologies/processes regarding the BDI model support [58].

Methodology	Belief	Desire (Goal)	Intention	Not support
KAOS Extension [78]		✓	✓	
JAAMAS [1]		✓		
Patrizia Ribino [65]	✓	✓	✓	
AGSIRA [53]		✓		
GORMAS [6]		✓		
ATABGE [70]		✓		
RE4Gaia [12]		✓		
Xu Huiying [44]		✓	✓	
REG for AOSE [32]		✓		
Extension GAIA [68]		✓		
B-Tropos [19]		✓		
JONGWON LEE [50]		✓		
Prabhat Ranjan [66]		✓		
SRAMO [52]		✓		
Zhiqi Shen [74]		✓		
SONIA [2]				✓
BDI ASP [46]	✓	✓	✓	
Tropos [62]	✓	✓	✓	
Cysneiros [57]	✓	✓	✓	
Chiung-Hui [24]		✓		
KAOS [8]		✓		
PRACTIONIST [60]	✓	✓	✓	
Murray [61]				✓
ASPECS [25]		✓		
REF [18]		✓		
Sen and Jain [72]		✓		
Hsieh et al. [43]				✓
Sutcliffe [77]		✓		
Agile Sen and Jain [71]		✓		
CREWS-EVE [39]		✓		
Cao et al. [55]		✓		
HOMER [80]		✓		
Wu et al. [81]		✓		
Liu and Li [54]				✓
Mahmoud et al. [38]				✓
Ashamalla et al. [7]		✓		
Consensus [13]				✓
Hilaire et al. [42]		✓		
Gaur and Soni [33]		✓		
Passos et al. [64]		✓		
PLANT [79]		✓		
Ronald et al. [69]		✓		
aMIAC [30]				✓
GAIA [23]		✓		
PASSI [27]				✓
INGENIAS-SCRUM [35]				✓
ADELFE [14]				✓
O-MaSE [29]		✓	✓	
PROMETHEUS [63]	✓	✓	✓	
MESSAGE [20]		✓		
OSOAD [21]		✓		
COMOMAS [34]	✓	✓	✓	
MAS-COMMONKADS [45]	✓	✓	✓	
MASSIVE [51]				✓
Total	8	43	11	11

addition, the methodology uses what has been called “internal use cases” to represent plans, desires, and beliefs, however, no adaptation of the use-cases diagram to represent these concepts is presented, and this approach contains conceptual errors, such as the representation of agents as use-cases.

Tropos [62] supports the BDI model through the i^* framework. However, this framework presents limitations in the coverage of this model. These limitations can be perceived by the lack of a way to visualize when a goal becomes an intention and how the beliefs affect the agents’ goals. In addition to the support provided by i^* , Tropos dedicates, in the design phase, a stage to BDI agents, however we have not analysed this stage in depth because it belongs to the design phase and not to the requirements engineering phase.

Cysneiros methodology [57], theoretically supports the BDI model since it uses the i^* framework. However, although it uses a framework containing concepts of BDI agents, the work of Cysneiros does not state at any moment that it works with the BDI model.

PRACTIONIST [60] is a goal-oriented methodology. However, unlike the other goal-oriented methodologies found in this review, the authors of PRACTIONIST describe its goals as goals of BDI agents.

PRACTIONIST is based on the BDI model, presenting total support for its use. As an overview of the support to this model, we can highlight that the methodology cover the following concepts: a set of perceptors that hear some relevant external stimuli (perceptions); a set of beliefs that represent information the agent got about its internal state and the external environment; a set of goals that the agent desire to achieve, representing the states or activities the agent would like to perform in the system, related to the agent desires or intentions; a set of goals relation that the agent uses during the deliberation and reasoning means-end process; a set of plans that are the means to achieve the intentions; a set of actions the agent can execute to act in the environment; and a set of effectors that really execute the actions.

However, although PRACTIONIST structure is adequate to the BDI model use, the methodology has structural problems for its application. The main issue is that the methodology seems to be more suitable for the design area, the only point that contributed to include this methodology as suitable for requirements engineering is that the authors understand that the methodology is useful for requirements analysis. Even so, PRACTIONIST does not provide a lifecycle for its requirements engineering and does not detail how should be done the requirements analysis for BDI agents.

Prometheus [63] states to have full support for the BDI model, however, it does not present in its models examples or explanations of how to apply the BDI model, leaving open for interpretation by whoever applies the methodology. We considered it a risk, as it can generate inconsistencies and even errors during the system development.

Even so, some concepts can be understood through Prometheus models, such as the use of “data” to represent the beliefs. However, only data repositories are identified and not the beliefs. Belief repositories are linked to the goals, which may result in conflict since in a BDI agent view, beliefs contain the knowledge of an agent. Furthermore, although goals are represented in the Prometheus models, it is not clear how they change in intentions.

Lastly, regarding CoMoMAS [34] and MAS-CommonKADS [45] methodologies, we believe that they work with the same notation to represent the BDI model, since both methodologies have their models derived from the CommonKADS language, a conceptual language for modeling.

However, the MAS-CommonKADS methodology does not detail the use of the BDI model, it only states that they support this model. Regarding the CoMoMAS methodology, it uses the BDI model to represent the system internal structure, using the concepts of belief, goal, and intention (described through the plan model). Nevertheless, in spite of the concepts seeming to follow a logic line of BDI agents, there are conceptual errors in its approach. In this view, intentions do not appear to be originated from goals, being represented practically as new goals. Furthermore, the example presented does not detail how beliefs influence a goal to become an intention.

5.4 Gaps Found in This Review

In this subsection we will detail the gaps found in this review, aiming to answer the fourth research question: “Which are the existing gaps in the methodologies that support RE for MAS?”

We noticed that only three studies cover the validation sub-area in their RE cycle. It demonstrates that the majority of the methodologies do not care with this phase that is so important to the systems quality.

We also noticed that just one study covers the four sub-areas of RE in its cycle [14]. On the other hand, this study does not support the BDI model, what demonstrates a gap and the need of the proposition of a methodology containing a requirements engineering phase that supports the BDI model.

Regarding the BDI model coverage, we understand that the support to just 8 studies from a total of 54 is a low number. Moreover, just two methodologies have as their focus to cover this model ([46,65]) and none of them cover elicitation and validation, what highlights a gap in the RE for MAS area.

Other point that we could identify as a neglect is that, among the methodologies that support BDI, only the Tropos methodology [62] covers the requirements elicitation and just the methodology proposed by Cysneiros [57] includes requirements validation. It demonstrates that most of the methodologies that support BDI focus on the requirements analysis and specification and that, besides these areas, there is space to be explored in the elicitation and validation areas.

We highlight in Table 7, which are the gaps of each study that fully supports the BDI model. Although some studies partially support this model, such as allowing the goals representation, they do not refer to the BDI model explicitly and we are only interested in studies that fully support the features needed for the BDI model.

As can be seen in Table 7, of all the studies fully supporting the BDI model, only the work of Cysneiros [57] covers the validation subarea, however it does not present any specific technique for check the requirements. In this methodology it is mentioned only that the artifacts should be validated, but without defining how this should be done. It demonstrates that the validation subarea, so important for quality control, is being neglected by the current methodologies.

Table 7. Studies that fully support the BDI model and its gaps.

Methodology	Gaps
Patrizia Ribino [65]	<ul style="list-style-type: none"> ● Does not cover the Elicitation, Analysis, and Validation subareas ● Does not allow specifying a communication pattern ● Does not allow specifying communication between agents or agents and external users ● Does not allow specifying in a clear way the goals associated with the agents ● Does not allow identifying when a goal becomes an intention ● Does not use a requirements specification standard
BDI ASP [46]	<ul style="list-style-type: none"> ● Does not cover the Elicitation, Specification, and Validation subareas ● Does not allow specifying when a goal becomes an intention ● The UML adaptation contains conceptual errors, such as applying use cases to represent agents ● Represents intentions and plans as synonyms
Tropos [62]	<ul style="list-style-type: none"> ● Does not cover the validation subarea ● Does not present any elicitation technique ● Does not allow specifying a communication pattern ● Does not allow specifying the kind of message sent between agents or agents and external users ● Does not provide resources to identify and model perceptions associated with agents ● Does not present any requirements elicitation technique ● Does not use any requirements specification standard ● Does not allow specifying when a goal becomes an intention
Cysneiros and Yu [57]	<ul style="list-style-type: none"> ● Does not present any requirements validation technique ● Does not allow specifying a communication pattern ● Does not allow specifying communication between agents or agents and external users ● Does not cover the elicitation subarea ● Does not use any requirements specification standard
PRACTIONIST [60]	<ul style="list-style-type: none"> ● Does not cover the Elicitation and Validation subareas ● Does not allow specifying a communication pattern ● Does not allow specifying communication between agents or agents and external users ● Does not use any requirements specification standard ● Does not concretely present the requirements engineering subareas.
Prometheus [63]	<ul style="list-style-type: none"> ● Does not cover the Elicitation, Specification, and Validation subareas ● Does not show how to apply the BDI model in its structure ● Does not allow specifying when a goal becomes an intention
CoMoMAS [34]	<ul style="list-style-type: none"> ● Does not cover the Elicitation, Specification, and Validation subareas ● Does not allow specifying when a goal becomes an intention
MAS-CommonKADS [45]	<ul style="list-style-type: none"> ● Does not cover the Elicitation and Validation subareas ● Does not allow specifying a communication pattern ● Does not allow specifying communication between agents or agents and external users ● Does not provide resources to identify and model perceptions associated with agents ● Does not use any requirements specification standard ● Does not allow specifying when a goal becomes an intention

Another point that can be highlighted is that from these studies only the Tropos [62] methodology covers the requirements elicitation subarea, however, Tropos does not present any technique for capturing specific requirements for multiagent systems, such as those necessary to apply the BDI model.

Furthermore, the studies presented by Ribino [65], Tropos [62], Cysneiros [57], PRACTIONIST [60], and MAS-CommonKADS [45] despite covering the requirements specification subarea, they do not adopt a specification standard. Whereas, according Alsanad [3] due to the general complexity of requirements specification documents, software stakeholders tend not to use them efficiently, therefore, according to Boyarchuk [16], during requirements formation and formulation, it is important to follow the standards that rule the software development process. Moreover, the adoption of a specification standard establishes what information about requirements must be collected and how this information must be organized and structured in documents.

6 Threats to Validity

During the planning and execution of this review, some factors were characterized as threats to the research validity. The potential threats are discussed to orient the interpretation of this work:

1. **Construct Validity:** The reliability of the search string defined to select relevant works can be a threat to the construct. To minimize this threat the string was calibrated with the execution of several tests and the area expert was consulted about the most used terms.
2. **Internal Validity:** A possible threat could have arisen from the individual interpretation of each researcher, something that could have led to the exclusion of relevant studies. To minimize this threat, the protocol of this review was strictly followed, considering mainly the inclusion and exclusion criteria. When necessary, a researcher with experience in this area was consulted to reach a consensus about the acceptance of the identified studies.
3. **External Validity:** Another possible threat is that some studies could not have been found because it does not contain keywords defined in the search string. To minimize this threat, the book “Handbook on Agent-Oriented Design Processes” [26] was used as research source and some classical papers were manually selected by a specialist in the area. To complement the research we performed a manual search in the methodologies found aiming to ensure the use of studies with the most recent version.
4. **Coverage Validity:** Regarding the possible papers that were not captured by our String, we intend, as a future work, to apply the snowballing technique trying to find more relevant papers. Another issue is that the snowballing technique can allow us to find more papers about the analysed methodologies, since in this analysis we focused only on the last paper of each methodology and this practice may not fully guarantee a complete coverage of the methodology.
5. **Conclusion Validity:** In spite of following a systematic protocol, systematic reviews are subject to human error, especially in the data extraction from papers. To mitigate this threat, the data extraction was performed by two independent researchers following the strategy defined in Subsect. 4.7 and, in case of divergences, a specialist in the area was consulted.

7 Insights for Future Works

Based on this systematic literature review, we found some gaps that could become future research. Among these gaps, we highlight two of them that are already being addressed in our research group: I) the need to propose requirements engineering processes for the context of multi-agent systems that cover the four sub-areas of requirements engineering and support the resources needed to the BDI model, as none of the 54 processes analyzed in this review actually support these characteristics, which shows a good research opportunity; and II) the need to propose techniques related to the four sub-areas of requirements engineering, especially where the main gaps were found, such as requirements elicitation, specification and validation.

Thus, in the elicitation subarea, we believe it is necessary to develop techniques to obtain requirements related to the BDI model which we have not found. Regarding direct contact with the stakeholders, we found only Homer [80] technique addressing this issue, however Homer presents limitations that need to be addressed in future works.

Regarding the studies that support the specification subarea, none of them adopt a specification standard that establishes how the requirements needed for the BDI model must be organized and documented. Therefore, it is necessary to propose or adopt a standard to document the collected and analysed requirements. Furthermore, this standard should allow describing not only the requirements associated with agents, but also the requirements describing functionalities accessed by normal users.

Finally, these documents must be verified using appropriate techniques to ensure their quality. However, none of the studies returned in our systematic review presented a validation subarea in a consistent way and none of them presented a specific technique for verifying the documented requirements.

8 Conclusions and Future Works

Our original systematic review published in [58], aimed to answer research questions about which methodologies for multiagent systems support the requirements engineering lifecycle, what is the level of coverage of the requirements engineering subareas of these methodologies and which of them have focus on the BDI model. Our study has categorized the studies directly related to the research theme and revealed new possibilities of scientific research in this area.

In that work, we presented 54 studies that support at least one of the subareas of requirements engineering, somehow adapted or applied in the multiagent systems context. Among these studies we noticed that only 8 methodologies support the features needed to the BDI model. However, we have not discussed in depth these methodologies due to lack of space.

Thus, this new study aimed to detail the support of these methodologies regarding the BDI model, as well as to discuss the gaps found in these studies. Regarding the elicitation subarea, we noticed that only the Homer methodology [80] presented a new way to elicit requirements from the direct contact with the stakeholders. However Homer does not support requirements related to the BDI model. Therefore, we realized the need of proposing a technique to elicit requirements related to systems based on the BDI model.

Another gap found is related to the requirements specification subarea, considering that no study uses a specification standard. Lastly, analysing the studies supporting the BDI model, we noticed that only the work of Cysneiros [57] covers the validation subarea, but it does not present any specific technique for requirements verification. Thus, we concluded that further studies are need to create a requirements specification standard for multi-agent systems, as well as, to develop a verification technique to ensure that the requirements documented in this standard are correct, consistent and non-ambiguous.

Based on the data synthesis from the retrieved studies and their gaps, we are currently proposing a requirements engineering process for multiagent systems covering

the subareas of elicitation, analysis, specification, and validation. Furthermore, this process is based on the notation proposed in the Multi-Agent Systems Requirements Modeling Language (MASRML) [37] that aims to support the BDI model.

This requirement engineering process will contain a elicitation technique and a verification technique both specific for multiagent systems based on the BDI model. Moreover, the requirements specification will follow an internationally recognized standard extended to the multiagent systems context. All these works are already in development in our research group.

References

1. Abushark, Y., Miller, T., Thangarajah, J., Winikoff, M., Harland, J.: Requirements specification via activity diagrams for agent-based systems. *Auton. Agents Multi-Agent Syst.* **31**(3), 423–468 (2016). <https://doi.org/10.1007/s10458-016-9327-7>
2. Alonso, F., Frutos, S., Martínez, L., Montes, C.: SONIA: a methodology for natural agent development. In: Gleizes, M.-P., Omicini, A., Zambonelli, F. (eds.) *ESAW 2004*. LNCS (LNAI), vol. 3451, pp. 245–260. Springer, Heidelberg (2005). https://doi.org/10.1007/11423355_18
3. AlSanad, A., Chikh, A.: Reengineering of software requirement specification. In: Rocha, Á., Correia, A.M., Tan, F.B., Stroetmann, K.A. (eds.) *New Perspectives in Information Systems and Technologies, Volume 2*. AISC, vol. 276, pp. 95–111. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-05948-8_10
4. Alzetta, F., Giorgini, P., Marinoni, M., Calvaresi, D.: RT-BDI: a real-time BDI model. In: Demazeau, Y., Holvoet, T., Corchado, J.M., Costantini, S. (eds.) *PAAMS 2020*. LNCS (LNAI), vol. 12092, pp. 16–29. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-49778-1_2
5. Alzetta, F., Giorgini, P., Najjar, A., Schumacher, M.I., Calvaresi, D.: In-Time explainability in multi-agent systems: challenges, opportunities, and roadmap. In: Calvaresi, D., Najjar, A., Winikoff, M., Främling, K. (eds.) *EXTRAAMAS 2020*. LNCS (LNAI), vol. 12175, pp. 39–53. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-51924-7_3
6. Argente, E., Botti, V., Julian, V.: GORMAS: an organizational-oriented methodological guideline for open MAS. In: Gleizes, M.-P., Gomez-Sanz, J.J. (eds.) *AOSE 2009*. LNCS, vol. 6038, pp. 32–47. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19208-1_3
7. Ashamalla, A., Beydoun, G., Low, G.: Model driven approach for real-time requirement analysis of multi-agent systems. *Comput. Lang. Syst. Struct.* **50**, 127–139 (2017)
8. Banach, R.: A deidealisation semantics for KAOS. In: *Proceedings of the 2010 ACM Symposium on Applied Computing, SAC '10*, pp. 267–274. Association for Computing Machinery, New York, NY, USA (2010). <https://doi.org/10.1145/1774088.1774146>
9. Berenbach, B., Paulish, D., Kazmeier, J., Rudorfer, A.: *Software & Systems Requirements Engineering: In Practice*. McGraw-Hill Inc, New York (2009). 978-0-07-160548-9
10. Biolchini, J., Mian, P.G., Natali, A.C.C., Travassos, G.H.: *Systematic review in software engineering* (2005)
11. Blanes, D., Insfran, E., Abrahão, S.: Requirements engineering in the development of multi-agent systems: a systematic review. In: Corchado, E., Yin, H. (eds.) *IDEAL 2009*. LNCS, vol. 5788, pp. 510–517. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04394-9_62

12. Blanes, D., Insfran, E., Abrahão, S.: RE4Gaia: a requirements modeling approach for the development of multi-agent systems. In: Ślęzak, D., Kim, T., Kiumi, A., Jiang, T., Verner, J., Abrahão, S. (eds.) ASEA 2009. CCIS, vol. 59, pp. 245–252. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-10619-4_30
13. Bokma, A., Slade, A., Kerridge, S., Johnson, K.: Engineering large-scale agent-based systems with consensus. *Robot. Comput. Integr. Manuf.* **11**(2), 81–90 (1994)
14. Bonjean, N., Meftteh, W., Gleizes, M.P., Maurel, C., Migeon, F.: ADELFE 2.0. In: Cossentino, M., Hilaire, V., Molesini, A., Seidita, V. (eds.) *Handbook on Agent-Oriented Design Processes*, pp. 19–63. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-39975-6_3
15. Bourque, P., Fairley, R.E., et al.: *Guide to the software engineering body of knowledge (SWEBOOK (R)): Version 3.0*. IEEE Computer Society Press (2014). 978-0-7695-5166-1
16. Boyarchuk, A., Pavlova, O., Bodnar, M., Lopatto, I.: Approach to the analysis of software requirements specification on its structure correctness. In: *CEUR Workshop Proceedings 2623*, pp. 85–95 (2020). <http://ceur-ws.org/Vol-2623/paper9.pdf>
17. Bratman, M., et al.: *Intention, Plans, and Practical Reason*, vol. 10. Harvard University Press Cambridge, MA (1987). 1–57586-192-5
18. Bresciani, P., Donzelli, P.: REF: a practical agent-based requirement engineering framework. In: Jeusfeld, M.A., Pastor, Ó. (eds.) *ER 2003*. LNCS, vol. 2814, pp. 217–228. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-39597-3_21
19. Bryl, V., Mello, P., Montali, M., Torroni, P., Zannone, N.: B-tropos: agent-oriented requirements engineering meets computational logic for declarative business process modeling and verification. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5056, pp. 157–176 (2008). <http://dx.doi.org/10.1007/978-3-540-88833-8-9>
20. Caire, G., Coulier, W., Garijo, F., Gómez-Sanz, J., Pavón, J., Kearney, P., Massonet, P.: The Message Methodology. In: Bergenti, F., Gleizes, MP., Zambonelli, F. (eds) *Methodologies and Software Engineering for Agent Systems. Multiagent Systems, Artificial Societies, and Simulated Organizations*, vol. 11, pp. 177–194. Springer, US, Boston, MA (2004). https://doi.org/10.1007/1-4020-8058-1_12
21. Cao, L.: OSOAD methodology. In: *Metasynthetic Computing and Engineering of Complex Systems*. AIKP, pp. 111–129. Springer, London (2015). https://doi.org/10.1007/978-1-4471-6551-4_6
22. Cernuzzi, L., Cossentino, M., Zambonelli, F.: Process models for agent-based development. *Eng. Appl. Artif. Intell.* **18**(2), 205–222 (2005)
23. Cernuzzi, L., Molesini, A., Omicini, A.: The gaia methodology process. In: Cossentino, M., Hilaire, V., Molesini, A., Seidita, V. (eds.) *Handbook on Agent-Oriented Design Processes*, pp. 141–172. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-39975-6_6
24. Lee, C.H.L., Liu, A.: A method for agent-based system requirements analysis. In: *2002 Proceedings of the Fourth International Symposium on Multimedia Software Engineering*, pp. 214–221. IEEE (2002). <https://doi.org/10.1109/MMSE.2002.1181615>
25. Cossentino, M., Galland, S., Gaud, N., Hilaire, V., Koukam, A.: A glimpse of the ASPECS process documented with the FIPA DPDF template. In: *Proceedings of The Multi-Agent Logics, Languages, and Organisations Federated Workshops (MALLOW 2010)*, vol. 627 (2010)
26. Cossentino, M., Hilaire, V., Molesini, A., Seidita, V. (eds.): *Handbook on Agent-Oriented Design Processes*. Springer, Heidelberg (2014). <https://doi.org/10.1007/978-3-642-39975-6>
27. Cossentino, M., Seidita, V.: PASSI: process for agent societies specification and implementation. In: Cossentino, M., Hilaire, V., Molesini, A., Seidita, V. (eds.) *Handbook on Agent-Oriented Design Processes*, pp. 287–329. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-39975-6_10

28. De Silva, L., Meneguzzi, F., Logan, B.: BDI agent architectures: a survey. In: IJCAI International Joint Conference on Artificial Intelligence (2020)
29. DeLoach, S.A., Garcia-Ojeda, J.C.: The o-mase methodology. In: Cossentino, M., Hilaire, V., Molesini, A., Seidita, V. (eds.) *Handbook on Agent-Oriented Design Processes*, pp. 253–285. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-39975-6_9
30. Domann, J., Hartmann, S., Burkhardt, M., Barge, A., Albayrak, S.: An agile method for multiagent software engineering. *Procedia Computer Science*, vol. 32, pp. 928–934 (2014). <https://doi.org/10.1016/j.procs.2014.05.513>, <http://www.sciencedirect.com/science/article/pii/S1877050914007133>. In: *The 5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014), the 4th International Conference on Sustainable Energy Information Technology (SEIT-2014)*, Elsevier
31. Dorri, A., Kanhere, S.S., Jurdak, R.: Multi-agent systems: a survey. *IEEE Access*, vol. 6, pp. 28573–28593. Institute of Electrical and Electronics Engineers Inc (2018)
32. Fuentes-Fernández, R., Gómez-Sanz, J., Pavón, J.: Requirements elicitation and analysis of multiagent systems using activity theory. *Systems, Man and Cybernetics, Part A: Systems and Humans*, *IEEE Transactions on* vol. 39, pp. 282–298. IEEE (2009). <https://doi.org/10.1109/TSMCA.2008.2010747>
33. Gaur, V., Soni, A.: A novel approach to explore inter agent dependencies from user requirements. *Procedia Technology*, vol. 1, pp. 412–419. Elsevier (2012). <https://doi.org/10.1016/j.protcy.2012.02.093>
34. Glaser, N.: The CoMoMAS methodology and environment for multi-agent system development. In: Zhang, C., Lukose, D. (eds.) *DAI 1996*. LNCS, vol. 1286, pp. 1–16. Springer, Heidelberg (1997). <https://doi.org/10.1007/BFb0030078>
35. González-Moreno, J.C., Gómez-Rodríguez, A., Fuentes-Fernández, R., Ramos-Valcárcel, D.: Ingenias-scrum. In: Cossentino, M., Hilaire, V., Molesini, A., Seidita, V. (eds.) *Handbook on Agent-Oriented Design Processes*, pp. 219–251. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-39975-6_8
36. Guedes, G.T.A., Vicari, R.M.: A UML profile oriented to the requirements modeling in intelligent tutoring systems projects. In: Bramer, M. (ed.) *IFIP AI 2010*. IAICT, vol. 331, pp. 133–142. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15286-3_13
37. Guedes, G., Souza Filho, I., Gaedicke, L., Mendonça, G., Vicari, R., Brusius, C.: Masrml - a domain-specific modeling language for multi-agent systems requirements. *Int. J. Soft. Eng. Appl. (IJSEA)* **11**(5) (2020)
38. Hajer, B.M., Taieb, B.R., Raouf, K.: A new mas based approach modeling the QMS continual improvement. In: *2009 IEEE International Conference on Systems, Man and Cybernetics*, pp. 4734–4739. IEEE (2009)
39. Haumer, P., Heymans, P., Jarke, M., Pohl, K.: Bridging the gap between past and future in re: a scenario-based approach. In: *Proceedings IEEE International Symposium on Requirements Engineering (Cat. No.PR00188)*, pp. 66–73. IEEE (1999)
40. Henderson-Sellers, B., Gorton, I.: Agent-based software development methodologies. In: *White Paper, Summary of Workshop at the OOPSLA*, vol. 2003 (2002)
41. Herzig, A., Lorini, E., Perrussel, L., Xiao, Z.: BDI logics for BDI Architectures: old problems, new perspectives. *KI - Künstliche Intelligenz* **31**(1), 73–83 (2016). <https://doi.org/10.1007/s13218-016-0457-5>
42. Hilaire, V., Cossentino, M., Gechter, F., Rodriguez, S., Koukam, A.: An approach for the integration of swarm intelligence in mas: an engineering perspective. *Expert Systems with Applications*, vol. 40, pp. 1–24. Elsevier (2012). <https://doi.org/10.1016/j.eswa.2012.08.058>
43. Hsieh, M., Hung, W., Shin, S., Lin, S., Huang, T.: Spoken dialogue agent interface requirements modeling based on PASSI methodology. In: *2008 Eighth International Conference on Intelligent Systems Design and Applications*, vol. 1, pp. 339–342. IEEE (2008)

44. Huiying, X., Zhi, J.: An agent-oriented requirement graphic symbol representation and formalization modeling method. In: SETN 2002: Methods and Applications of Artificial Intelligence, vol. 4, pp. 569–574. IEEE Computer Society (2009). <https://doi.org/10.1109/CSIE.2009.923>
45. Iglesias, C.A., Garijo, M., González, J.C., Velasco, J.R.: Analysis and design of multi-agent systems using MAS-CommonKADS. In: Singh, M.P., Rao, A., Wooldridge, M.J. (eds.) ATAL 1997. LNCS, vol. 1365, pp. 313–327. Springer, Heidelberg (1998). <https://doi.org/10.1007/BFb0026768>
46. Jo, C.H., Einhorn, J.: A BDI agent-based software process. *J. Object Technol.* **4**, 101–121 (2005). <https://doi.org/10.5381/jot.2005.4.9.a3>
47. Kitchenham, B., Charters, S.: Guidelines for performing systematic literature reviews in software engineering (2007)
48. Kitchenham, B., Brereton, P.: A systematic review of systematic review process research in software engineering. *Inf. Softw. Technol.* **55**(12), 2049–2075 (2013)
49. Larsen, J.B.: Agent programming languages and logics in agent-based simulation. In: Sieminski, A., Kozierkiewicz, A., Nunez, M., Ha, Q.T. (eds.) Modern Approaches for Intelligent Information and Database Systems. SCI, vol. 769, pp. 517–526. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-76081-0_44
50. Lee, J., Lee, H.: Strategic agent based web system development methodology. *Int. J. Inf. Technol. Decis. Making* **7**, 309–337. World Scientific (2008). <https://doi.org/10.1142/S02196220080029837>,
51. Lind, J. (ed.): MASSIVE views. In: Iterative Software Engineering for Multiagent Systems. LNCS (LNAI), vol. 1994, pp. 121–204. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45162-5_5
52. Lindoso, A., Girardi, R.: The SRAMO technique for analysis and reuse of requirements in multi-agent application engineering. In: 9th Workshop on Requirements Engineering, pp. 41–50. Workshop on Requirements Engineering (2006)
53. Liu, L., Jin, Z., Lu, R., Yang, H.: Agent-oriented requirements analysis from scenarios. In: O’Shea, J., Nguyen, N.T., Crockett, K., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2011. LNCS (LNAI), vol. 6682, pp. 394–405. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22000-5_41
54. Liu, W., Li, M.: Requirements planning with event calculus for runtime self-adaptive system. In: 2015 IEEE 39th Annual Computer Software and Applications Conference, vol. 2, pp. 77–82. IEEE (2015)
55. Cao, L., Zhang, C., Luo, D., Chen, W., Zamani, N.: Integrative early requirements analysis for agent-based systems. In: Fourth International Conference on Hybrid Intelligent Systems (HIS’04), pp. 118–123. IEEE (2004)
56. Mahela, O.P., et al.: Comprehensive overview of multi-agent systems for controlling smart grids. *CSEE J. Power Energy Syst.* **8**(1), 115–131 (2020)
57. Marcio Cysneiros, L., Yu, E.: Requirements engineering for large-scale multi-agent systems. In: Garcia, A., Lucena, C., Zambonelli, F., Omicini, A., Castro, J. (eds.) SELMAS 2002. LNCS, vol. 2603, pp. 39–56. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-35828-5_3
58. Mendonça, G., Souza, Filho, I., Guedes, G.: A systematic review about requirements engineering processes for multi-agent systems. In: 13th International Conference on Agents and Artificial Intelligence (ICAART 2021), vol. 1, pp. 69–79 (2021)
59. Mendonça, G.D., de Souza Filho, I.P., Guedes, G.T.A.: A systematic review about requirements engineering processes for multi-agent systems. In: ICAART, vol. 1, pp. 69–79 (2021)
60. Morreale, V., Bonura, S., Francaviglia, G., Centineo, F., Cossentino, M., Gaglio, S.: Goal-oriented development of bdi agents: the practionist approach. In: Proceedings of the 2006

- IEEE/WIC/ACM International Conference on Intelligent Agent Technology, pp. 66–72. IEEE (2006). <https://doi.org/10.1109/IAT.2006.71>
61. Murray, J.: Specifying agent behaviors with UML statecharts and statedit. In: Polani, D., Browning, B., Bonarini, A., Yoshida, K. (eds.) RoboCup 2003. LNCS (LNAI), vol. 3020, pp. 145–156. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-25940-4_13
 62. Mylopoulos, J., Castro, J., Kolp, M.: The evolution of tropos. In: Seminal Contributions to Information Systems Engineering, pp. 281–287. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36926-1_22
 63. Padgham, L., Thangarajah, J., Winikoff, M.: Prometheus research directions. In: Shehory, O., Sturm, A. (eds.) Agent-Oriented Software Engineering, pp. 155–171. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-54432-3_8
 64. Passos, L.S., Rossetti, R.J., Gabriel, J.: An agent methodology for processes, the environment, and services. In: Rossetti, R.J., Liu, R. (eds.) Advances in Artificial Transportation Systems and Simulation, pp. 37–53. Academic Press, Boston (2015). <https://doi.org/10.1016/B978-0-12-397041-1.00003-0>, <http://www.sciencedirect.com/science/article/pii/B9780123970411000030>
 65. Patrizia, R., Lodato, C., Sabatucci, L.: Ontology and goal model in designing BDI multi-agent systems. In: CEUR Workshop Proceedings, vol. 1099 (2013)
 66. Ranjan, P., Misra, A.: A novel approach of requirements analysis for agent based system. In: Seventh ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD'06), pp. 299–304. IEEE (2006). <https://doi.org/10.1109/SNPD-SAWN.2006.5>
 67. Rao, A.S., Georgeff, M.P.: Bdi agents: From theory to practice. In: Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95), pp. 312–319 (1995)
 68. Rodriguez, L., Hume, A., Cernuzzi, L., Insfran, E.: Improving the quality of agent-based systems: integration of requirements modeling into gaia. In: 2009 Ninth International Conference on Quality Software, pp. 278–283. IEEE Computer Society (08 2009). <https://doi.org/10.1109/QSIC.2009.43>
 69. Ronald, N., Dignum, V., Jonker, C., Arentze, T., Timmermans, H.: On the engineering of agent-based simulations of social activities with social networks. Information and Software Technology, **54**(6), 625–638 (2012). <https://doi.org/10.1016/j.infsof.2011.12.004>, <http://www.sciencedirect.com/science/article/pii/S0950584911002485>. Special Section: Engineering Complex Software Systems through Multi-Agent Systems and Simulation, Elsevier
 70. Sen, A., Hemachandran, K.: Elicitation of goals in requirements engineering using agile methods. In: 2010 IEEE 34th Annual Computer Software and Applications Conference Workshops, pp. 263–268. IEEE (2010). <https://doi.org/10.1109/COMPSACW.2010.53>
 71. Sen, A.M., Jain, S.K.: An agile technique for agent based goal refinement to elicit soft goals in goal oriented requirements engineering. In: 15th International Conference on Advanced Computing and Communications (ADCOM 2007), pp. 41–47. IEEE (2007)
 72. Sen, A.M., Jain, S.K.: A visualization technique for agent based goal refinement to elicit soft goals in goal oriented requirements engineering. In: Second International Workshop on Requirements Engineering Visualization (REV 2007), pp. 2–2. IEEE (2007). <https://doi.org/10.1109/REV.2007.1>
 73. Shah, U.S., Jinwala, D.C.: Resolving ambiguities in natural language software requirements: a comprehensive survey. ACM SIGSOFT Softw. Eng. Notes **40**(5), 1–7 (2015)
 74. Shen, Z., Li, D., Miao, C., Gay, R., Miao, Y.: Goal-oriented methodology for agent system development. In: IEICE TRANSACTIONS on Information and Systems, vol. 89, pp. 95–101. The Institute of Electronics, Information and Communication Engineers (2005). <https://doi.org/10.1109/IAT.2005.80>

75. Singh, D., Padgham, L., Logan, B.: Integrating BDI agents with agent-based simulation platforms. *Autonomous Agents and Multi-Agent Systems* **30**(6), 1050–1071 (2016). <https://doi.org/10.1007/s10458-016-9332-x>
76. Sudeikat, J., Braubach, L., Pokahr, A., Lamersdorf, W.: Evaluation of agent-oriented software methodologies – examination of the Gap between modeling and platform. In: Odell, J., Giorgini, P., Müller, J.P. (eds.) *AOSE 2004*. LNCS, vol. 3382, pp. 126–141. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-30578-1_9
77. Sutcliffe, A.: Requirements engineering for complex collaborative systems. In: *Proceedings Fifth IEEE International Symposium on Requirements Engineering*, pp. 110–117. IEEE (2001). <https://doi.org/10.1109/ISRE.2001.948550>
78. Ulfat-Bunyadi, N., Mohammadi, N., Heisel, M.: Supporting the systematic goal refinement in kaos using the six-variable model. In: *13th International Conference on Software Technologies*, pp. 102–111. SciTePress (2018). <https://doi.org/10.5220/0006850701020111>
79. Wang, Y., Zhao, L., Wang, X., Yang, X., Supakkul, S.: Plant: A pattern language for transforming scenarios into requirements models. *Int. J. Hum. Comput. Stud.* **71**, 1026–1043. Elsevier (2013). <https://doi.org/10.1016/j.ijhcs.2013.08.001>
80. Wilmann, D., Sterling, L.: Guiding agent-oriented requirements elicitation: Homer. In: *Fifth International Conference on Quality Software (QSIC'05)*, pp. 419–424. IEEE (2005)
81. Wu, H., Liu, L., Ma, W.: Optimizing requirements elicitation with an i* and bayesian network integrated modelling approach. In: *IEEE 34th Annual Computer Software and Applications Conference Workshops*, pp. 182–188. IEEE (2010)