Revisión Sistemática de la Literatura de las Extensiones de JADE para la Programación de Ecosistemas Inteligentes Basados en Agentes

Pablo Pico-Valencia[1,*], **Juan A. Holgado-Terriza**[2]

[1] *Tecnologías de la Información, Pontificia Universidad Católica del Ecuador, Esmeraldas, Ecuador*
[2] *Departamento de Lenguajes y Sistemas Informáticos, Universidad de Granada, Granada, Spain*

Fecha de recepción: 21 de junio de 2022 — Fecha de aceptación: 18 de noviembre de 2023

Resumen

Los agentes de software son entidades fundamentales en el desarrollo de sistemas distribuidos reactivos inteligentes. Para el desarrollo de este tipo de entidades se han propuesto varias herramientas como lenguajes de programación, marcos de trabajo y plataformas. Sin embargo, pocas de estas herramientas han sido efectivamente adoptadas y mantenidas. La herramienta JADE es considerada como el marco de trabajo más prometedor entre los programadores de sistemas multiagente y lleva muchos años evolucionando. Por ello, en este trabajo se realiza una revisión sistemática de la literatura cuyo objetivo es identificar las principales propuestas de extensión de JADE desarrolladas para la creación de sistemas inteligentes basados en agentes en distintos dominios. Se evidenció que las principales áreas en las que se extendió el marco JADE fueron: el comercio electrónico, la inteligencia ambiental, el Internet de las cosas, la industria y redes inteligentes. Asimismo, se identificó que las principales características soportadas dentro de los marcos extendidos propuestos estaban orientadas principalmente a cubrir aspectos no contemplados por JADE, tales como: tiempo real, componentes deliberativos, razonamiento semántico, orientado a servicios, colaboración, flujos de trabajo y normas.

Palabras Clave: Sistema multiagente; agente de software; JADE; sistema inteligente; marco de trabajo.

A Systematic Literature Review on JADE Extensions for Programming Agent-Based Intelligent Ecosystems

Abstract

Software agents are important entities in the development of intelligent and reactive distributed systems. For the development of these types of entities, several tools have been proposed, such as programming languages, frameworks, and platforms. However, few of these tools have been effectively adopted and maintained. The JADE tool is the most promising framework among programmers of multi-agent systems, and it has been evolving for many years. Therefore, in this paper, a systematic literature review aimed at identifying the main proposals to extend JADE for the development of intelligent agent-based systems in different domains has been addressed. We have found that the main areas in which the JADE framework was extended were e-commerce, ambient intelligence, the Internet of Things, industry, and smart grid. Likewise, it was identified that the main features supported within the extended proposed frameworks were mainly oriented to cover aspects not covered by JADE, such as real-time, deliberative components, semantic reasoning, service-oriented architecture, collaboration, workflows, and norms.

Keywords: Multi-agent system; software agent; JADE; intelligent systems; framework.

*pablo.pico@pucese.edu.ec

Nota: Este artículo de divulgación es parte de Ingeniería–Revista Académica de la Facultad de Ingeniería, Universidad Autónoma de Yucatán, Vol. 28, No. 2, 2024, ISSN: 2448-8364

1. Introduction

Agent-oriented technologies arise from a need in the field of robotics to assist in the creation of reactive systems capable of autonomously executing tasks based on changes that occur in their environment. In this sense, software agents, the basic unit of the agent-oriented paradigm [1], have so far been an alternative for the development of proactive, adaptive, socio-collaborative, and intelligent systems in different domains [2].

There is no consensus on how to define an agent. Jennings and Wooldridge (1998) [3] have probably provided the most widely adopted definition in the literature: "An agent is an encapsulated computer system that is situated in some environment and is capable of acting flexibly and autonomously within this environment to achieve its design goals". A set of software agents constitutes a multi-agent system (MAS) [2]. A MAS can be comprised of a group of agents based on reactive, deliberative, or hybrid architecture [4]. MASs are suitable for addressing distributed problems, where characteristics such as decentralization, autonomy, and coordination play a central role.

According to the adopted architecture, agents can execute simple or more complex tasks. For example, in the case of reactive agents, they generally drive their behavior based on a trigger-event mechanism that executes a set of If-Then rules [5]. On the other hand, in the case of deliberative agents, they perform tasks based on human mental abilities such as beliefs, desires, and intentions [6]. In the latter case, deliberative agents can exhibit a level of intelligence comparable to that commonly seen in reactive agents. Finally, to model more complex behaviors that leverage the benefits of the two agent architectures mentioned above, it is possible to model MASs using a hybrid architecture [7].

For the creation of reactive, deliberative, and hybrid agents, specialized software tools have been developed. These technological tools consist of programming languages (e.g., CLAIM, FLUX, MINERVA, KABUL, DALI, ReSpecT, AL, 3APL, Jason, IMPACT, Go, AF-APL) and frameworks or platforms (e.g., TuCSon, Zeus, JADE, PADE, SPADE, Jadex, and DESIRE) [8–10]. Most of these tools have been analyzed, compared, and evaluated from various perspectives such as performance and communication to determine their scope and limitations in terms of intelligence, collaboration, proactivity, mobility, adaptation, domains of use (e.g., industry, healthcare, entertainment), and emerging support technologies (e.g., services, resources, ontologies) that enhance the capabilities of MASs. segundo autor

To date, there are numerous tools available for creating MASs. However, only a few of the proposed tools have gained significant popularity and widespread adoption among MAS developers. There are several reasons why the use of these tools hasn't reached a high level of popularity and usage. One of the reasons is that they fail to comply with the communication standards proposed by the Foundation for Intelligent Physical Agents (FIPA) [11,12]. Additionally, some of these tools are not distributed as open-source software, not available online, or simply not designed for general-purpose use. In many cases, the existing mechanisms for agent development are tailored to specific research group goals within the field of agents.

Despite the fact that many of the proposed tools have not met developers' expectations, there is a widely used tool called the Java Agent DEvelopment Framework (JADE) [13]. JADE is a framework specifically designed for agent development using Java. This tool supports the communication standards proposed by FIPA, such as ACL (Agent Communication Language), which enables agents to interoperate between different platforms [14]. Additionally, JADE provides a set of interaction protocols, an agent mobility model, and an ontology-based communications model that greatly facilitate the development of intelligent systems.

This research aims to understand the current situation regarding how the JADE framework has been extended to support the development of intelligent systems that the official version of JADE is not capable of. Similarly, the study has identified the main application domains of extended JADE versions in the development of intelligent systems so that developers can understand their scope in terms of development, applicability, and deployment of IT production systems.

This paper is organized as follows: Section 2 describes the methodology followed in conducting the systematic literature review process. Section 3 analyzes and describes the results obtained. This section also addresses the questions raised in the preliminary phase of the research. Finally, in Section 4, the main conclusions and future directions for the development of agents using the extended versions of the JADE framework are presented.

2. Methodology

The methodological guidelines proposed by Kofod-Petersen [15] were applied to address this study. These guidelines are useful for conducting systematic literature reviews in the field of Computer Science as they recommend specialized data sources in this area. Furthermore, the proposed methodology offers guidance on formulating the scientific research equation, defining research questions, and selecting the primary studies to be analyzed.

Definition of research questions (RQs)

Since the JADE framework is currently the most popular and widely used tool for agent development [16], it has been chosen as the focus of this study to identify its extended versions. Six research questions have been strategically formulated to gather information about the main frameworks derived from the extended versions of JADE that have been published between 2010 and 2021. Additionally, the analysis examines how these new versions of JADE have been utilized to address specific problems that are not supported by the official JADE version. The research questions (RQ1-RQ6) are defined as follows:

- **RQ1**. What kind of JADE-based frameworks has been introduced for developing intelligent systems?, What is their main goal?
- **RQ2**. Which features have been empowered or introduced into the extended JADE-based frameworks?
- **RQ3**. Are the new versions of JADE sufficiently documented?
- **RQ4**. What domains or problems have been addressed using the extended JADE versions?
- **RQ5**. What intelligent mechanisms have been enhanced or introduced by the JADE-extended frameworks?
- **RQ6**. What emerging technologies have been employed by the JADE-extended frameworks?

Selection of information sources

The previously defined research questions have provided the foundation for identifying a set of terms that have guided our scientific search process. The primary terms consist of the words (“`jade`”, “`java agent development framework`”), and (“`exten*`”). Based on these terms, we have strategically formulated a search string (SS) following the structure described in (1):

$$((\text{“}jade\text{”}\ OR\ \text{“}java\ development\ agent\ framework\text{”})\ AND\ (\text{“}exten*\text{”})) \quad (1)$$

Once the search string was defined, we proceeded to identify the information sources to execute the search query and retrieve the most relevant studies in which new versions of JADE have been published in high-impact scientific publications. Following the methodology applied for literature reviews, a total of six primary information sources (PIS) were considered. These information sources include three specialized digital libraries in the field of Computer Science (PIS1 to PIS3), two documentary databases (PIS1 and PIS5), and a web academic search engine (PIS6).

- **PIS1:** Institute of Electrical and Electronics Engineers (IEEE) Digital Library, available at: `http://ieeexplore.ieee.org/`
- **PIS2:** Association for Computing Machinery (ACM) Digital Library, available electronically at: `https://dl.acm.org/`
- **PIS3:** Wiley Online Lybrary, available at: `http://www.onlinelibrary.wiley.com/`
- **PIS4:** Web of Science Database, available at: `https://webofknowledge.com/`
- **PIS5:** Scopus Database, available at: `https://www.scopus.com/`
- **PIS6:** Google Scholar, available at: `https://scholar.google.com/`

Selection of studies

The search process was conducted in February 2021 using each of the specified search engines, and it focused on the titles of publications. This approach allowed us to exclude papers that were not relevant to the study topic and avoid duplicate entries. It is important to emphasize that the intention of this study was not to analyze MASs where JADE is applied, but rather to analyze frameworks and middleware that have been extended from JADE to incorporate additional functionalities not provided by the official version of JADE.

Language (English), date of publication (2002-2021), and the focus on studies where JADE has been improved in its official version were considered as inclusion criteria. After analyzing the titles and abstracts of the articles, a total of 30 scientific studies (S1-S30) were selected, including 6 published in journals, 24 published in proceedings and books sponsored by international conferences, and 1 technical report. All of the retrieved studies, as shown in Table 1, were written in English, published within the specified time frame, and accessible through the information sources used for retrieval. Only study S27 was partially accessible, with only its abstract available.

3. Results and discussion

After reading the full texts of the 30 selected articles listed in Table 1, it became evident that the proposals for extended frameworks based on JADE were published between 2002 and 2021. As depicted in Figure 1, it was

Table 1: Primary studies selected to be analized in the literature review

#	Source	Year	Type of study	Country	Reference
S1	PIS5	2014	Conference	Italy	[17]
S2	PIS5	2006	Conference	Canada	[18]
S3	PIS5	2010	Conference	China	[19]
S4	PIS5	2011	Conference	Germany	[20]
S5	PIS5	2009	Journal	Italy	[21]
S6	PIS5	2010	Conference	Spain	[22]
S7	PIS5	2011	Conference	Croatia	[23]
S8	PIS6	2002	Conference	Australia	[24, 25]
S9	PIS2	2010	Journal	Iran	[26]
S10	PIS4	2002	Conference	USA	[27]
S11	PIS3	2002	Journal	Brazil	[28]
S12	PIS6	2002	Conference	Brazil	[29]
S13	PIS6	2002	Journal	Iran	[30, 31]
S14	PIS6	2008	Conference	Italy	[32]
S15	PIS6	2014	Conference	Iraq	[33]
S16	PIS6	2005	Conference	Germany	[34]
S17	PIS6	2010	Journal	Italy	[16]
S18	PIS6	2017	Journal	Italy	[35]
S19	PIS6	2004	Conference	Canada	[36]
S20	PIS6	2014	Conference	Czech Republic	[37]
S21	PIS6	2013	Conference	Italy	[38]
S22	PIS6	2015	Conference	Portugal	[39]
S23	PIS6	2007	Conference	Australia	[40]
S24	PIS6	2016	Technical report	USA	[41]
S25	PIS6	2011	Conference	Brazil	[42]
S26	PIS6	2006	Conference	Italy	[43]
S27	PIS6	2019	Conference	Algeria	[44]
S28	PIS6	2016	Conference	Poland	[45]
S29	PIS6	2008	Conference	Italy	[46]
S30	PIS6	2007	Conference	China	[47]

also observed that the highest number of studies was proposed in 2002 and 2010. However, in the other years, several proposals were also published, with the exceptions of 2003, 2012, and 2018. These data demonstrate that JADE continues to be employed for creating specialized frameworks for agent development, utilizing emerging technologies and novel approaches.

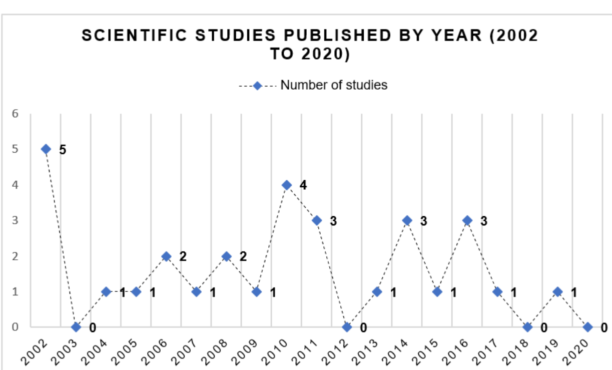

Figure 1: Analized studies grouped by year

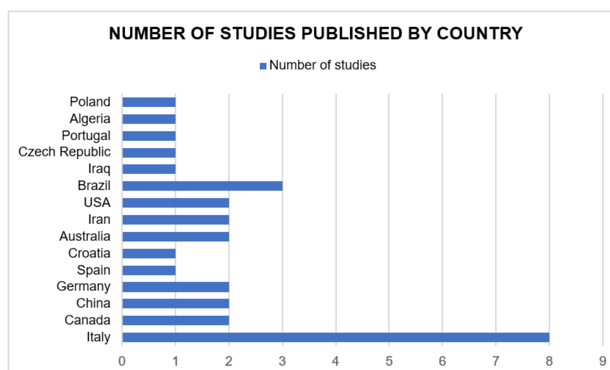

Figure 2: Analized studies grouped by country of publication

It was also observed that the analyzed frameworks were proposed by researchers from various countries. Figure 2 positions Italy as the country with the most research on improving the JADE framework, with 8 proposals. This is presumably due to the fact that JADE was created by Italian researchers. Additionally, countries such as Brazil, the USA, Iran, Australia, Germany, China, and Canada have also made innovative proposals to introduce new concepts and technologies into JADE, with 2 proposals each. Finally, other countries, including Poland, Algeria, Portugal, Iraq, the Czech Republic, Croatia, and Spain, have contributed with one proposal each to improve the JADE framework.

In addition to the overall results described above, the analysis of the full text of the 30 retrieved studies provided insights into the type of study conducted by each researcher. Seventy percent of all the studies did not include a case study as a validation test for the introduced model, which could be a significant limitation for developers intending to implement a system using such frameworks. On the other hand, the remaining thirty percent of the studies addressed similar aspects to the first group but included an experimentation process that involved quantitative measurements, such as communication performance (S2 [18]), simulation steps (S4 [20]), rejection rate in access control tasks (S5 [21]), percentage of successful service executions (S22 [39]), and message exchange time for varying message sizes (S2 [18]).

The analysis of the retrieved studies also enabled us to address each of the research questions formulated in this study (RQ1-RQ6). A summary of the results obtained, along with comparative tables and analysis from the researchers'perspective, is presented below.

3.1. What kind of JADE-based frameworks has been introduced for developing intelligent systems?, What is their main goal? (RQ1)

The investigation has revealed the existence of 30 frameworks and middleware for the development of MASs based on the JADE agent programming framework. Firstly, as illustrated in Table 2, frameworks focused on improving the development of distributed systems have been identified. The following proposals are included in this line: S1 [17], S2 [18], S8 [24, 25], S17 [16], S19 [36], S20 [37], S24 [41] and S30 [47]. Their main objective is to introduce technologies such as JXTA (Juxtapose) and web services to establish communication with heterogeneous devices connected to a network, regardless of the network topology. This enables the distribution of agents'tasks and their interoperability with external services and modern applications distributed over the Internet.

On the other hand, frameworks were also identified to facilitate the modeling of tasks that JADE agents must execute (behavior). Several studies focused on this aspect, aiming to simplify the modeling of behaviors for JADE agents. One approach to achieve this goal involved the use of workflows (S17 [16], S29 [46]). Another approach aimed to enhance the generic behavior model implemented by JADE, making it easier to model complex tasks and adapt them to the agents' operating environment and user requirements (S7 [23], S10 [27], S23 [40]).

In the realm of modeling agent tasks, frameworks have been identified to facilitate task modeling based on time constraints (S11 [28]) and priority (S6 [22]). These frameworks enable the creation of soft real-time systems where tasks are scheduled using algorithms other than the First In, First Out (FIFO) approach used in the official version of JADE. Examples of such algorithms include Last in, First Out (LIFO), Earliest Deadline First (EDF), Preemptive EDF (PEDF), Deadline Monotonic (DM), Preemptive Deadline Monotonic (PDM), Priority-Based Scheduling (PRIO), Preemptive Priority-Based Scheduling (PPRIO), Shortest Job First (SJF), and Shortest Remaining Time First (SRTF) (S11 [28]).

Complementary to these efforts, a proposal was identified that facilitates the automatic generation of MASs in JADE (S18 [35]). This study focused on a model-driven approach, where the MAS modeler only needs to specify abstract data to create the meta-model and automatically generate the code that implements the MASs.

Frameworks focused on the creation of groups and organizations of JADE agents were also identified. The creation of societies in the MAS domain is an aspect not implemented in the official version of JADE. Hence, studies have proposed the introduction of concepts such as teams (S28 [45]), agent organizations, and roles. These frameworks enable the modeling of collaborative behaviors between agents (S14 [32], S16 [34]), with special attention given to coordination, negotiation, and interaction mechanisms (S4 [20], S9 [26]).

Other frameworks have focused on the security aspect of agents. In one study (S5 [21]), security was implemented at the agent level using biometric authentication mechanisms to control access to the functionalities provided by the agents in a MAS. Additionally, an agent reputation mechanism was introduced, allowing actions requested from agents to be based on their trust level regarding previously executed tasks. This aspect is crucial in systems, particularly in critical systems such as smart grid systems (S1 [17]). Another study (S15 [33]) addressed the security issue in MASs through an ontology that stores metadata related to agent access, enabling access control to system operations for each agent or user.

As evidenced in the previous cases, ontologies have been applied in the agent domain. These proposed frameworks are supported by ontologies, which introduce semantic annotations (also referred to as knowledge or information) to enable more complex reasoning. Ontologies help achieve a higher level of intelligence than what is supported by JADE alone. A clear example is the first study (S3 [19]) that utilized semantic annotations to improve agent negotiation processes. Similarly, the second study (S21 [38]) enabled agents to carry out automatic discovery processes of smart objects that have published

Table 2: Main JADE-based frameworks for developing intelligent systems introduced in the literature

#	Approach	Goal
S1	Smart Grid JADE [17]	Smart grid JADE-based systems.
S2	JADE-JXTA [18]	Better agent-based distributed systems integrating JXTA.
S3	JADE Protocol Ont. [19]	Interaction protocol according the agent environment.
S4	JREP [20]	Cooperation at micro- and marco-level.
S5	JADE-S [21]	Biometric autentication for secure accessing to JADE agents.
S6	JADE AmI [22]	Ambient Intelligent JADE-based systems.
S7	JADE JBehaviorTree [23]	Composition of agent tasks based on a behaviour tree model.
S8	WS2JADE [24, 25]	Integration og JADE with web services.
S9	Holo-JADE [26]	Holonic JADE-based systems.
S10	Smart Agent JADE [27]	Improving the JADE agent behaviour concept.
S11	RT-JADE [28]	Real-time schedulling with preemtive mobile agents.
S12	JAMDER [29]	Scaling the JADE reactive architecture to a BDI architecture.
S13	ACE-JADE [30, 31]	Combining agent processes with autonomic computing.
S14	PowerJADE [32]	Managing organizations and roles in JADE systems.
S15	OJADEAC [33]	Agent access control model based on ontologies.
S16	JADEOrgs [34]	Adding organization and roles in JADE systems at runtime.
S17	JADE Semantic P2P [16]	Interoperability with P2P and server systems
S18	JADEL [35]	Model driven JADE MAS development.
S19	IG-JADE-PKSlib [36]	Web service composition and provisioning into JADE agents.
S20	JADE-JBossESB [37]	Communication with enterprise services from JADE.
S21	JADE-SO [38]	Discovering smart objects into ecosystems of devices.
S22	SAJaS [39]	Simulation and development agent systems from JADE.
S23	JADE-FSM [40]	Adaptation according running enviroment and user needs.
S24	Blue-JADE [41]	Interoperation of JADE with modern applications servers.
S25	BDI4JADE [42]	Suporting JADE agent with the BDI architecture.
S26	J-ALINAs [43]	Optimizing the JADE agents communications.
S27	NorJADE [44]	Normative JADE agent systems.
S28	JADE-A-Team [45]	Forming teams of JADE agents.
S29	WADE [46]	Workflows for the execution of tasks in JADE.
S30	JADE-EE [47]	Large scale enterprise JADE systems.

and indexed their services in the yellow page catalog implemented by JADE. Other proposals have utilized JADE's own ontology model or ontologies in OWL format to introduce knowledge and enable agents to make more accurate decisions aimed at achieving their goals (e.g., S8 [24, 25], S9 [26], S12 [29], S14 [32], S15 [33], S16 [34], S17 [16], S26 [43], S27 [44], S28 [45]). The utilization of these proposals depends on their specific requirements. However, in the analyzed frameworks, ontologies have been leveraged to model intelligent tasks such as automatic service discovery, access control, automatic task composition, coordination of agent organizations, and agent interaction.

Finally, frameworks were also identified to develop MASs based on integrating a normative model, supporting a deliberative architecture, and creating simulation environments. Firstly, the normative model (S27 [44]) enabled JADE agents to reason within the constraints of a set of rules that establish control over these entities. Secondly, in terms of supporting deliberative components (S12 [29], S25 [42]), two studies proposed the integration of an additional layer on JADE, enabling developers to handle BDI components, namely beliefs, desires, and intentions. As a result, JADE agents were able to model their goals based on the concept of plans. Thirdly, a framework for modeling real-world behaviors in simulation environments (S22 [39]) was evident, which was compatible with JADE agents exhibiting behaviors such as those seen in ants or swarms.

3.2. Which features have been empowered or introduced into the extended JADE-based frameworks? (RQ2)

The previous research question described how various framework proposals have made efforts to enhance and introduce new features and functionalities into JADE, thereby expanding its range of applicable domains. Table 3 presents 20 technical features that have been improved or introduced in JADE. Among these features, 9 (adaptability, interoperability, mobility, scalability, code reusing, scheduling policy, semantic, modularization, flexibility) have been enhanced to optimize the already implemented features in the official version of JADE. The remaining 11 features (security, authorization, reputation, cooperation, regulation, BDI, maintainability, service-supported, task composition, task distribution, and collaboration) represent innovative aspects, models, and architectures introduced to enhance JADE's potential for the development of intelligent systems.

The introduction of these extended features enhances the applicability of JADE in a broader range of scenarios. Some of these enhanced features include ontology-based communication protocols, improved finite state machine (FSM) behavior, interoperability with third-party services and systems that do not use the FIPA-ACL language, creation of ontology-based environments, real-time algorithms for scheduling agent tasks, and adaptive agents capable of adapting themselves according to changes in the environment.

On the other hand, we can highlight the introduction of novel features, such as the implementation of biometric identification and authorization, mechanisms for executing the services offered by agents, the management of reputation mechanisms that define trust in an ecosystem, the addition of deliberative elements that bring intelligence closer to human intelligence, the creation of agent societies regulated by a normative model, and the utilization of RESTful web services for task execution distribution.

3.3. Were extended versions of JADE framework sufficiently documented in online format? (RQ3)

The results of the study regarding the documentation availability and source code accessibility of the proposed frameworks are presented in Table 4. These findings indicate that despite the researchers' efforts to extend the JADE tool to support emerging technologies and functionalities not covered in its official standard, most of the proposed tools have inadequate documentation. The documentation available for the new frameworks is limited to the technical-scientific descriptions provided in each publication indexed in the digital libraries and bibliographic databases explored in this study.

Overall, 90 % of the frameworks analyzed in this study were found to have poor documentation. Through web exploration, it was determined that only 10 % of the analyzed frameworks had sufficient documentation to guide developers in learning and applying them to real-world problems. In addition to the indexed scientific publications, these 10 % of frameworks (S11 [28], S16 [34], S22 [39], S24 [41], S25 [42], S27 [44], S29 [46]) had technical documents and digital guides available on a website or repository.

3.4. What kind of domains or specific problems were solved using the extended JADE versions? (RQ4)

The domains or application areas emphasized in the analyzed frameworks are presented in Table 5. The table also provides insight into the specific problems that the proposed frameworks have been employed to solve. In summary, the researchers have directed their efforts towards adapting the JADE tool to specific domains such as industry, ubiquitous spaces, and marketing. As discussed in RQ2, the frameworks have been extended to support technical features including task distribution, web services support, deliberative intelligence support, semantics support, and agent societies support. However, in this section, we will examine the practical impact and benefits of these frameworks.

In practical terms, the analyzed frameworks have demonstrated different orientations and impacts on the agent-based approach. Among the most significant directions are proposals focused on the development of intelligent negotiation systems, industrial control systems, pervasive systems, and general systems that exemplify non-real-life scenarios.

Regarding negotiation systems in marketing, the proposals have aimed at establishing coordination mechanisms to model negotiation processes more accurately between agents. In this regard, one study combined semantic web technologies to enhance the agents' knowledge (strategy library) and enable effective communication even in the presence of environmental changes. The agents learned to adapt and conducted intelligent negotiations in e-commerce settings (S3 [19]). Similarly, another study (S4 [20]) proposed a collaborative system for an airport with autonomous transportation services, incorporating macro/micro interaction levels. Based on the information published by the transport agents at the macro level, the agents at the micro level negotiated services such as pick-up/drop-off positions and schedules, which were then offered to the passengers. Furthermore, in the field of negotiation, other proposals implemented mechanisms for agent collaboration and negotiation in travel systems and book sales. For instance, in the case of book purchases, an agent sought out sellers, inquired about the price of the desired item, and ultimately selected the seller with the best offer (S18 [35]). A similar situation arose in the context of flight ticket purchases, where a customer considered offerings from multiple companies (S19 [36]).

On the other hand, in the context of automatic control and decision-making systems in industrial environments, the proposed frameworks placed greater emphasis on aspects such as security. One of the frameworks aimed at the development of smart grids (S1 [17]), enabling developers to manage authentication and authorization

Table 3: Empowered and introduced features into the JADE-extended frameworks

#	Approach	1 Security	2 Autorization	3 Reputation	4 Adaptability	5 Interoperability	6 Mobility	7 Cooperation	8 Scalability	9 Regulation	10 Code reusing	11 Scheduling policy	12 Semantic	13 Modularization	14 BDI	15 Maintainability	16 Service-supported	17 Flexibility	18 Task composition	19 Task distribution	20 Collaboration
S1	Smart Grid JADE [17]	**Yes**	**Yes**	No	No	**Yes**	**Yes**	No	No	No	No	No	No	No	No	No	No	**Yes**	**Yes**	No	No
S2	JADE-JXTA [18]	No	No	No	No	No	No	No	**Yes**	No	No	No	No	No	No	No	No	No	No	**Yes**	No
S3	JADE Protocol Ontology [19]	No	No	No	**Yes**	No	No	No	No	No	No	No	**Yes**	No	No	No	No	No	No	No	**Yes**
S4	JREP [20]	No	No	No	No	No	No	**Yes**	**Yes**	No	No	**Yes**	No	No	No	No	No	**Yes**	No	No	**Yes**
S5	JADE-S [21]	**Yes**	No	**Yes**	No	No	No	No	**Yes**	No	No	No	No	No	No	No	No	No	No	No	No
S6	JADE AmI [22]	No	No	No	No	No	No	No	No	No	**Yes**	**Yes**	No	**Yes**	No	No	No	No	No	**Yes**	No
S7	JADE JBehaviorTree [23]	No	No	No	**Yes**	No	No	**Yes**	No	No	**Yes**	**Yes**	No	**Yes**	No	No	No	No	**Yes**	No	No
S8	WS2JADE [24, 25]	No	No	No	No	No	No	No	No	No	**Yes**	No	**Yes**	No	No	No	No	**Yes**	**Yes**	**Yes**	No
S9	Holo-JADE [26]	No	No	No	No	No	No	**Yes**	No	No	No	No	**Yes**	No	No	No	No	No	No	**Yes**	**Yes**
S10	Smart Agent JADE [27]	No	No	No	**Yes**	No	No	No	No	No	No	No	No	No	No	No	No	**Yes**	No	No	No
S11	RT-JADE [28]	No	No	No	No	No	**Yes**	No	No	No	No	**Yes**	No	No	No	No	No	No	No	No	No
S12	JAMDER [29]	No	No	No	No	No	No	No	No	No	No	No	**Yes**	No	**Yes**	No	No	No	No	No	**Yes**
S13	ACE-JADE [30, 31]	No	No	No	**Yes**	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No
S14	PowerJADE [32]	No	No	No	No	No	No	No	No	No	No	No	**Yes**	No	**Yes**	No	No	No	No	No	**Yes**
S15	OJADEAC [33]	**Yes**	**Yes**	**Yes**	No	**Yes**	No	No	No	No	No	No	**Yes**	No	No	No	No	No	No	No	No
S16	JADEOrgs [34]	No	No	No	**Yes**	No	No	No	**Yes**	**Yes**	No	No	**Yes**	**Yes**	No	No	No	**Yes**	No	No	**Yes**
S17	JADE Semantic P2P [16]	No	No	No	No	**Yes**	No	No	**Yes**	No	No	No	**Yes**	No	No	No	**Yes**	No	No	**Yes**	No
S18	JADEL [35]	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No
S19	IG-JADE-PKSlib [36]	No	No	No	No	**Yes**	No	No	No	No	No	No	No	No	No	No	**Yes**	No	No	**Yes**	No
S20	JADE-JBossESB [37]	No	No	No	**Yes**	**Yes**	No	No	No	No	No	**Yes**	No	No	No	No	No	**Yes**	No	No	No
S21	JADE-SO [38]	No	No	No	No	No	No	No	No	No	No	No	**Yes**	No	No	No	**Yes**	No	No	No	No
S22	SAJaS [39]	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	**Yes**	No
S23	JADE-FSM [40]	No	No	No	**Yes**	No	No	No	No	No	No	No	No	No	No	No	No	**Yes**	**Yes**	No	No
S24	Blue-JADE [41]	No	No	No	No	**Yes**	No	No	**Yes**	No	No	No	No	No	No	No	**Yes**	No	No	**Yes**	No
S25	BDI4JADE [42]	No	No	No	No	No	No	No	No	No	No	No	No	**Yes**	**Yes**	No	No	**Yes**	No	No	No
S26	J-ALINAs [43]	No	No	No	No	No	No	No	No	No	No	No	**Yes**	No	No	No	No	**Yes**	No	No	**Yes**
S27	NorJADE [44]	No	No	No	No	No	No	No	No	**Yes**	No	No	**Yes**	No	No	No	No	No	No	No	No
S28	JADE-A-Team [45]	No	No	No	**Yes**	No	No	No	**Yes**	No	No	No	**Yes**	No	No	No	No	**Yes**	No	No	**Yes**
S29	WADE [46]	No	No	No	No	**Yes**	No	No	**Yes**	No	No	No	No	No	No	**Yes**	No	**Yes**	No	No	No
S30	JADE-EE [47]	No	No	No	No	No	No	No	**Yes**	No	No	No	No	No	No	**Yes**	No	**Yes**	No	No	**Yes**

before agents could communicate and perform requested actions. These mechanisms were implemented using digital certificates associated with each system agent. However, a specific case study was not provided. Similarly, in sensitive systems like banking, biometric mechanisms were employed for agent access to prevent fraud (S5 [21]). Additionally, it was observed that industrial applications such as nanotechnology and aircraft manufacturing focused on two aspects: providing flexible adaptation mechanisms to address disturbances that may occur in processes and optimizing resource planning (S20 [37], S13 [30, 31]).

Furthermore, in the context of pervasive systems, the main focus has been on intelligent management in Ambient Intelligence environments. For instance, one framework aimed at maintaining thermal comfort in smart homes based on weather conditions such as temperature and humidity (S6 [22]). This was achieved through the use of a sensor and actuator repository, combined with a publication/subscription mechanism that allowed agents to operate based on changing weather conditions. Similarly, another framework focused on assisting in the management of smart objects, enabling more accurate control actions in IoT environments. In this case, agents governing object behavior published their services in catalogs of smart objects (an extension of the JADE Directory Facilitator), allowing users or other agents to discover and interact with them to achieve their goals (S21 [38]).

Finally, it is important to note that several of the analyzed frameworks did not include case studies that were specifically adapted to real-life scenarios. Instead, these studies presented hypothetical situations tailored to their research objectives. Four studies, in particular, exemplify this approach. These studies focused on the following aspects: testing security mechanisms implemented through digital certificates (S1 [17]), peer-to-peer communication among nodes connected in different networks (S2 [18]), testing virtual agent organizations and their interactions based on roles and regulatory controls (S16 [34]), and ensuring the correct execution of a modeled workflow that required the dynamic discovery of web services to distribute agent tasks (S17 [16]).

3.5. What kind of intelligent mechanisms were empowered or introduced by the JADE-extended frameworks? (RQ5)

This research has identified that 77 % of the analyzed frameworks were aimed at developing intelligent systems. Table 6 illustrates that these frameworks encompassed various intelligent mechanisms, including deli-

Table 4: Availability of documentation and source code of the analized frameworks

#	Approach	Documentation	Code availability
S1	Smart Grid JADE [17]	Poor	No
S2	JADE-JXTA [18]	Poor	No
S3	JADE Protocol Ontology [19]	Poor	No
S4	JREP [20]	Poor	No
S5	JADE-S [21]	Poor	No
S6	JADE AmI [22]	Poor	No
S7	JADE JBehaviorTree [23]	Poor	No
S8	WS2JADE [24, 25]	Poor	No
S9	Holo-JADE [26]	Poor	No
S10	Smart Agent JADE [27]	Poor	No
S11	RT-JADE [28]	Poor	Yes
S12	JAMDER [29]	Poor	No
S13	ACE-JADE [30, 31]	Poor	No
S14	PowerJADE [32]	Poor	No
S15	OJADEAC [33]	Poor	No
S16	JADEOrgs [34]	Poor	Yes
S17	JADE Semantic P2P [16]	Poor	No
S18	JADEL [35]	Poor	No
S19	IG-JADE-PKSlib [36]	Poor	No
S20	JADE-JBossESB [37]	Poor	No
S21	JADE-SO [38]	Poor	No
S22	SAJaS [39]	Enough	Yes
S23	JADE-FSM [40]	Poor	No
S24	Blue-JADE [41]	Poor	Yes
S25	BDI4JADE [42]	Enough	Yes
S26	J-ALINAs [43]	Poor	No
S27	NorJADE [44]	Poor	Yes
S28	JADE-A-Team [45]	Poor	No
S29	WADE [46]	Enough	Yes
S30	JADE-EE [47]	Poor	No

berative intelligence, rule-based intelligence, intelligence based on semantic reasoning, and intelligence based on Artificial Intelligence techniques. Additionally, the applications of these mechanisms to develop intelligent behaviors were also identified. These behaviors encompassed smart service discovery, intelligent negotiation, and intelligent adaptation.

Regarding deliberative intelligence, three proposals were identified that aimed to model agent objectives using the concept of a plan, which is typical of deliberative architectures. Therefore, the frameworks in studies S12 [29], S14 [32], and S15 [33] added a layer to JADE, enabling developers to create deliberative components such as beliefs, desires, and intentions. These components allowed JADE agents to operate not only based on reactive models but also to act in a manner similar to humans.

Another extensively integrated mechanism in the studied frameworks was the use of ontologies. Utilizing ontologies to describe specific entities with a shared vocabulary enabled agents to engage in semantic reasoning processes. This allowed agents to make more precise decisions and execute actions based on the meaning of metadata, rather than solely relying on its syntax. The studies that emphasized this intelligence mechanism the most were S3 [19], S8 [24, 25], S15 [33], and S25 [42].

Furthermore, Artificial Intelligence techniques such as neural networks, fuzzy logic, and genetic algorithms were applied to a lesser extent (S5 [21]). However, only limited details were provided. It was also observed that 23 % of the studies did not focus on modeling intelligence mechanisms (S2 [18], S6 [22], S7 [23], S11 [28], S22 [39], S23 [40]).

Finally, the aforementioned intelligence mechanisms were applied to create intelligent behaviors. These behaviors utilized the additional knowledge or metadata integrated by agents to discover more suitable services, interact with the most appropriate agents, engage in negotiation with agents that provide better quality of service, and compose tasks based on the available services. This information supported the enhancement of reasoning processes based on rules, as implemented in the official version of JADE, as well as smart adaptation, intelligent automatic service discovery, and intelligent negotiation processes among agents in an agent society (S1 [17], S10 [27], S13 [30, 31], S18 [35], S19 [36],

Table 5: Application domains and solved problems with the proposed frameworks

#	Approach	Domain of application	Specific problem
S1	Smart Grid JADE [17]	Smart grid	None
S2	JADE-JXTA [18]	Distributed systems	None
S3	JADE Protocol Ontology [19]	E-Commerce	Negotiation
S4	JREP [20]	Simulation	Airport system
S5	JADE-S [21]	E-Banking	Biometry
S6	JADE AmI [22]	Ambient Intelligence	Home comfort
S7	JADE JBehaviorTree [23]	Distributed systems	Agent communication
S8	WS2JADE [24, 25]	Marketing	Google and Amazon
S9	Holo-JADE [26]	Holonic systems	Library
S10	Smart Agent JADE [27]	Distributed systems	Autonous vehicles
S11	RT-JADE [28]	Distributed systems	Real time
S12	JAMDER [29]	Suply chain	Trading agent competition
S13	ACE-JADE [30, 31]	Complex systems	Nano Technology Swarm
S14	PowerJADE [32]	Games	Palyers interaction
S15	OJADEAC [33]	Context-aware	Security
S16	JADEOrgs [34]	Collaborative systems	None
S17	JADE Semantic P2P [16]	Distributed systems	None
S18	JADEL [35]	Marketing	Book trading
S19	IG-JADE-PKSlib [36]	Marketing	Air travels
S20	JADE-JBossESB [37]	Industry	Aircraft manufacturing
S21	JADE-SO [38]	IoT	Smart objects
S22	SAJaS [39]	Simulation	Games
S23	JADE-FSM [40]	Distributed sytems	General example
S24	Blue-JADE [41]	Distributed systems	Servers accessing
S25	BDI4JADE [42]	Distributed systems	General example
S26	J-ALINAs [43]	Distributed system	Document searching
S27	NorJADE [44]	No accesible	No accesible
S28	JADE-A-Team [45]	Optimization	Traveling salesman problem
S29	WADE [46]	Marketing	Control process
S30	JADE-EE [47]	Complex systems	Natural disaster forecast

S20 [37]).

3.6. What emerging technologies were employed by the JADE-extended frameworks? (RQ6)

Table 7 presents the primary technologies utilized by researchers to extend the JADE framework. The identified technologies include tools for enabling P2P communications, ontologies and semantic reasoners, service- and resource-oriented tools, process modeling languages, and languages for meta-model description. It is worth noting that some studies were restricted to using the tools available in JADE and did not employ or mention the use of additional tools (S7 [23], S9 [26], S11 [28], S13 [30, 31], S23 [40], and S28 [45]).

Regarding the technologies used to enhance agent communication with distributed systems, the JXTA tool (S2 [18], S17 [16]) was employed, providing protocols for fundamental peer-to-peer networking functions. Additionally, technologies oriented towards creating Java-based web services (JAX-RPC) and tools for invoking RESTful services were utilized. Service-oriented technologies (S8 [24,25]) and RESTful resources (S21 [38]) were selected to be used alongside JADE due to their compatibility for implementing services in Java.

As for the use of semantic technologies, the analyzed frameworks incorporated ontologies. Some proposals utilized the classes provided by JADE to establish a vocabulary understood by the agents, simplifying the execution of behavior-regulating rules. Conversely, other proposals leveraged specialized ontological tools like the Web Ontology Language (OWL) (S3 [19], S8 [24, 25], S27 [44]). These ontologies were integrated into the agent models after being created using the Protege tool. Furthermore, these proposals employed semantic reasoners (e.g., RDF Data Query Language, RQDL; Semantic Query-Enhanced Web Rule Language, SQWRL; Jena) to conduct inference processes based on the metadata managed by the agents as part of their structural components (S3 [19], S15 [33]).

Finally, it is worth noting that other tools were used less frequently. These tools included specialized technologies for defining agent processes and workflows, such as Web Service Business Process Execution Language (WS-BPEL) and Multiagent System Modeling Langua-

Table 6: Main contributions of the proposed frameworks for the development of intelligent systems

#	Approach	BDI reassoning	rules-bas reassoning	Semantic reassoning	IA-based	Automatic discovery	Smart negotiation	Smart adaptation	No specified
S1	Smart Grid JADE [17]	No	**Yes**	No	No	No	No	No	No
S2	JADE-JXTA [18]	No	No	No	No	No	No	No	**Yes**
S3	JADE Protocol Ontology [19]	No	No	**Yes**	No	No	**Yes**	No	No
S4	JREP [20]	No	No	No	No	No	**Yes**	No	No
S5	JADE-S [21]	No	No	No	**Yes**	No	No	**Yes**	No
S6	JADE AmI [22]	No	No	No	No	No	No	No	**Yes**
S7	JADE JBehaviorTree [23]	No	No	No	No	No	No	No	**Yes**
S8	WS2JADE [24, 25]	No	No	**Yes**	No	No	No	No	No
S9	Holo-JADE [26]	No	No	No	No	No	No	No	No
S10	Smart Agent JADE [27]	No	**Yes**	No	No	No	No	No	No
S11	RT-JADE [28]	No	No	No	No	No	No	No	**Yes**
S12	JAMDER [29]	**Yes**	No	No	No	No	No	No	No
S13	ACE-JADE [30, 31]	No	**Yes**	No	No	No	No	**Yes**	No
S14	PowerJADE [32]	**Yes**	No	No	No	No	No	No	No
S15	OJADEAC [33]	No	No	**Yes**	No	**Yes**	No	No	No
S16	JADEOrgs [34]	No	No	No	No	**Yes**	No	No	No
S17	JADE Semantic P2P [16]	No	No	No	No	**Yes**	No	No	No
S18	JADEL [35]	No	**Yes**	No	No	No	No	No	No
S19	IG-JADE-PKSlib [36]	No	**Yes**	No	No	No	No	No	No
S20	JADE-JBossESB [37]	No	**Yes**	No	No	No	No	No	No
S21	JADE-SO [38]	No	No	No	No	**Yes**	No	No	No
S22	SAJaS [39]	No	No	No	No	No	No	No	**Yes**
S23	JADE-FSM [40]	No	No	No	No	No	No	No	**Yes**
S24	Blue-JADE [41]	No	No	No	No	No	No	No	No
S25	BDI4JADE [42]	**Yes**	No	No	No	No	No	No	No
S26	J-ALINAs [43]	No	No	No	No	No	**Yes**	No	No
S27	NorJADE [44]	No	No	No	No	No	No	No	No
S28	JADE-A-Team [45]	No	No	No	No	**Yes**	No	No	No
S29	WADE [46]	No	No	No	**Yes**	No	No	**Yes**	No
S30	JADE-EE [47]	No	No	No	No	No	No	No	**Yes**

ge (MAS-ML) (S12 [29], S17 [16]). Additionally, tools like Eclipse Modeling Framework (EMF) were employed to create meta-models using the Ecore language. These meta-models provide domain-specific languages (DSL) for modeling agents or related aspects. These models can be automatically transformed into source code, resulting in correct-by-construction MAS implementations (S16 [34]). The use of tools like Jess and Prolog was also identified, which focus on executing logical inference processes based on facts and rules, such as knowledge-based systems (S10 [27], S19 [36]).

4. Conclusions and future works

Today, Java continues to hold its position as one of the most widely used programming languages in the world for developing distributed systems. Within the field of agent-based development, Java remains the predominant choice among developers. Over the past two decades (2000-2021), a total of 30 frameworks compatible with the JADE agent programming model in Java have been introduced. These frameworks have placed emphasis on specific application domains such as industry, pervasive environments, and marketing systems, as well as addressing specific problems like trading systems, simulation environments, and comfort systems (Table 5). However, JADE has also demonstrated its evolution by gradually expanding into the realm of

Table 7: Main contributions of the proposed frameworks for the development of intelligent systems

#	Approach	BDI reassoning
S1	Smart Grid JADE [17]	JADE-S, Security framework for JADE.
S2	JADE-JXTA [18]	JXTA, Juxtapose.
S3	JADE Protocol Ontology [19]	OWL, Web Ontology Language, SQWRL, Semantic Query-Enhanced Web Rule Language.
S4	JREP [20]	Repast Symphony, Recursive Porous Agent Simulation Toolkit.
S5	JADE-S [21]	JADE-S, Security framework for JADE.
S6	JADE AmI [22]	XML, Extensible Markup Language.
S7	JADE JBehaviorTree [23]	Not specified.
S8	WS2JADE [24, 25]	OWL, Web Ontology Language, JAX-RPC, Java API for XML-based RPC.
S9	Holo-JADE [26]	Not specified.
S10	Smart Agent JADE [27]	Jess, Java Expert System Shell.
S11	RT-JADE [28]	Not specified
S12	JAMDER [29]	MAS-ML2.0, Multi-agent System Modeling Language
S13	ACE-JADE [30, 31]	Not specified
S14	PowerJADE [32]	PowerJava
S15	OJADEAC [33]	JADE-S, Security framework for JADE, Jena, Java framework for building Semantic Web RDQL, RDF Data Query Language
S16	JADEOrgs [34]	Eclipse Modelling Framework Ecore Language
S17	JADE Semantic P2P [16]	WS-BPEL, Web Services Business Process Execution Language BEJ, BPEL Enhanced JADE Agent, JXTA, Juxtapose.
S18	JADEL [35]	Xtext
S19	IG-JADE-PKSlib [36]	Prolog IndiGolog-JADE, Open Agent Architecture
S20	JADE-JBossESB [37]	JBossESB, JBoss Enterprise Service Bus.
S21	JADE-SO [38]	RESTFul Web Service, Jersey, Jackson, Lucene
S22	SAJaS [39]	Repast Simphony
S23	JADE-FSM [40]	Not specified.
S24	Blue-JADE [41]	HP's Application Server
S25	BDI4JADE [42]	XML, eXtensible Markup Language.
S26	J-ALINAs [43]	MAPLE is a plug-in for the Ontology Editor Protégé.
S27	NorJADE [44]	OWL, Web Ontology Language.
S28	JADE-A-Team [45]	No specified
S29	WADE [46]	XPDL, XML Process Definition Language
S30	JADE-EE [47]	J2EE, Java Enterprise Edition.

Python-based development environments. This is evident through the introduction of frameworks like PADE (Python Agent DEvelopment framework) [48] and SPADE (Smart Python multi-Agent Development Environment) [49]. These frameworks have adapted the JADE model to enable the development of Multi-Agent Systems (MASs) in Python, a language that currently dominates the field of Artificial Intelligence development. Looking ahead, future frameworks should focus on extending PADE or SPADE, allowing for the creation of intelligent applications that can be executed across server environments, web platforms, mobile applications, and embedded systems.

The analyzed frameworks have demonstrated their versatility in supporting not just one, but multiple technical features associated with MASs (Table 3). Many of these features have contributed to the enhancement of intelligence-related aspects. Existing intelligence mechanisms have been further developed, and new ones have been introduced into the JADE programming model (Table 6). However, since the majority of these studies were conducted in the last decade (2000-2010) (Table 1), it is crucial to foster innovation and incorporate emerging technologies that are widely used today (Table 7). Some potential innovations include: (i) agents utilizing machine learning techniques; (ii) agents capable of distributing their intelligence using Cloud Computing, Fog Computing, or Edge Computing; (iii) agents equipped with the ability to integrate, merge, and analyze large volumes of data from Big Data ecosystems; (iv) agents with varying degrees of intelligence that can adapt to the requirements of developers in emerging intelligent environments (e.g., smart cities, IoT, autonomous vehicles); and finally, (v) mobile agents that prioritize security and privacy, ensuring the execution of actions by reliable agents with a good reputation.

Referencias

[1] S. Franklin, A. Graesser, Is it an agent, or just a program?: A taxonomy for autonomous agents, in: International Workshop on Agent Theories, Architectures, and Languages, Springer, 1996, pp. 21–35.

[2] A. Dorri, S. S. Kanhere, R. Jurdak, Multi-agent systems: A survey, IEEE Access 6 (2018) 28573–28593. doi:10.1109/ACCESS.2018.2831228.

[3] N. R. Jennings, K. Sycara, M. Wooldridge, A roadmap of agent research and development, Autonomous agents and multi-agent systems 1 (1) (1998) 7–38.

[4] K. Chin, K. S. Gan, R. Alfred, P. Anthony, D. Lukose, Agent architecture: An overview, Trans. Sci. Technol. 1 (1) (2014) 18–35.

[5] M. B. Hoy, If this then that: An introduction to automated task services, Medical reference services quarterly 34 (1) (2015) 98–103.

[6] S. Bora, I. Cakirlar, Adaptive bdi architecture for multi-agent systems, Int. J. Eng. Sci. Appl. 1 (2) (2017) 52–57.

[7] R. Girardi, A. Leite, A survey on software agent architectures., IEEE Intelligent Informatics Bulletin 14 (1) (2013) 8–20.

[8] R. H. Bordini, L. Braubach, M. Dastani, A. El Fallah Seghrouchni, J. J. Gomez-Sanz, J. Leite, G. OH́are, A. Pokahr, A. Ricci, A survey of programming languages and platforms for multi-agent systems, Informatica 30 (10) (2015) 33–44.

[9] M. Dastani, A Survey of Multi-agent Programming Languages and Frameworks, Springer Berlin Heidelberg, 2014, pp. 213–233.

[10] A. Singh, D. Juneja, A. K. Sharma, Agent development toolkits, International Journal of Advancements in Technology 2 (1) (2011) 158–164.

[11] C. Georgousopoulos, O. F. Rana, An approach to conforming a mas into a fipa-compliant system, in: Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 2, ACM, 2002, pp. 968–975.

[12] F. Bellifemine, A. Poggi, Fipa-compliant agent infrastructures, in: Methodologies and Software Engineering for Agent Systems, Springer, 2004, pp. 259–272.

[13] F. Bellifemine, G. Caire, A. Poggi, G. Rimassa, Jade: A software framework for developing multi-agent applications. lessons learned, Information and Software Technology 50 (1) (2008) 10–21.

[14] M. T. Kone, A. Shimazu, T. Nakajima, The state of the art in agent communication languages, Knowledge and Information Systems 2 (3) (2000) 259–284.

[15] A. Kofod-Petersen, How to do a structured literature review in computer science, Ver. 0.1. October 1 (2012).

[16] A. Poggi, M. Tomaiuolo, Extending the jade framework for semantic peer-to-peer service based applications, in: Developing Advanced Web Services through P2P Computing and Autonomous Agents: Trends and Innovations, IGI Global, 2010, pp. 18–35.

[17] A. Poggi, M. Tomaiuolo, P. Turci, Extending jade for agent grid applications, Vol. 13, 2004, pp. 352–357.

[18] E. Chen, D. Sabaz, W. Gruver, Jade and jxta extensions for the implementation of distributed systems, Vol. 1, 2006, pp. 740–745.

[19] X. Jin, S.-J. Ji, Y.-Q. Liang, B.-Y. Zou, Protocol ontology and its application in extending jade, Vol. 12, 2010, pp. V12264–V12268.

[20] J. Görmer, G. Homoceanu, C. Mumme, M. Huhn, J. Müller, Jrep: Extending repast simphony for jade agent behavior components, Vol. 2, 2011, pp. 149–154.

[21] S. Vitabile, V. Conti, C. Militello, F. Sorbello, An extended jade-s based framework for developing secure multi-agent systems, Computer Standards and Interfaces 31 (5) (2009) 913–930.

[22] A. Paz-Lopez, G. Varela, S. Vazquez-Rodriguez, J. Becerra, R. Duro, Some issues and extensions of jade to cope with multi-agent operation in the context of ambient intelligence, Advances in Intelligent Systems and Computing 71 (2010) 607–614.

[23] I. Bojic, T. Lipic, M. Kusek, G. Jezic, Extending the jade agent behaviour model with jbehaviourtrees framework, in: Proceedings of the 5th KES International Conference on Agent and Multi-agent Systems: Technologies and Applications, KES-AMSTA'11, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 159–168.

[24] X. T. Nguyen, R. Kowalczyk, M. B. Chhetri, A. Grant, Ws2jade: A tool for run-time deployment and control of web services as jade agent services, in: R. Unland, M. Calisti, M. Klusch (Eds.), Software Agent-Based Applications, Platforms and Development Kits, Birkhäuser Basel, Basel, 2005, pp. 223–251.

[25] X. T. Nguyen, R. Kowalczyk, Ws2jade: Integrating web service with jade agents, in: International Workshop on Service-Oriented Computing: Agents, Semantics, and Engineering, Springer, 2007, pp. 147–159.

[26] A. Esmaeili, N. Mozayani, Holojade: A role based holonic extension for jade, Int. J. Agent Technol. Syst. 2 (2) (2010) 49–63.

[27] M. Griss, S. Fonseca, D. Cowan, R. Kessler, Smartagent: Extending the jade agent behavior model, Proceedings of SEMAS (2002) 1–10.

[28] T. P. Filgueiras, L. M. Rodrigues, L. d. O. Rech, L. M. S. de Souza, H. V. Netto, Rt-jade: A preemptive real-time scheduling middleware for mobile agents, Concurrency and Computation: Practice and Experience (2019) e5061.

[29] Y. S. Lopes, E. J. Gonçalves, M. I. Cortés, S. Emmanuel, Extending jade framework to support different internal architectures of agents, in: 9th European Workshop on Multi-agent Systems (EUMAS 2011), 2011, pp. 1–15.

[30] A. Farahani, E. Nazemi, G. Cabri, Ace-jade: Autonomic computing enabled jade, in: 2016 IEEE 1st International Workshops on Foundations and Applications of Self* Systems (FAS* W), IEEE, 2016, pp. 267–268.

[31] A. Farahani, E. Nazemi, G. Cabri, N. Capodieci, Enabling autonomic computing support for the jade agent platform, Scalable Computing: Practice and Experience 18 (1) (2017) 91–103.

[32] M. Baldoni, G. Boella, M. Dorni, R. Grenna, A. Mugnaini, powerjade: Organizations and roles as primitives in the jade framework, In of WOA 2008: Dagli oggetti agli agenti, Evoluzione dellágent development: metodologie, tool, piattaforme e linguaggi (2008) 84–92.

[33] B. S. Mustafa, N. Aldabagh, Ojadeac: An ontology based access control model for jade platform, Editorial Preface 5 (5) (2014) 42.

[34] C. Madrigal-Mora, K. Fischer, Adding organisations and roles to jade with jadeorgs, in: Agent-Based Technologies and Applications for Enterprise Interoperability, Springer, 2005, pp. 98–117.

[35] F. Bergenti, E. Iotti, S. Monica, A. Poggi, Agent-oriented model-driven development for jade with the jadel programming language, Computer Languages, Systems & Structures 50 (2017) 142–158.

[36] E. Martínez, Y. Lespérance, IG-JADE-PKSlib: An Agent-Based Framework for Advanced Web Service Composition and Provisioning, in: Proceedings of the Workshop on Web Services and Agent-Based Engineering (AAMAS-2004), 2004, pp. 2–10.

[37] P. Vrba, M. Fuksa, M. Klíma, Jade-jbossesb gateway: Integration of multi-agent system with enterprise service bus, in: 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, 2014, pp. 3663–3668.

[38] G. Fortino, M. Lackovic, W. Russo, P. Trunfio, A discovery service for smart objects over an agent-based middleware, in: International Conference on Internet and Distributed Computing Systems, Springer, 2013, pp. 281–293.

[39] H. L. Cardoso, Sajas: enabling jade-based simulations, in: Transactions on Computational Collective Intelligence XX, Springer, 2015, pp. 158–178.

[40] S. K. Goh, M. B. Chhetri, R. Kowalczyk, Jade-fsm-engine: A deployment tool for flexible agent behaviours in jade, in: Proceedings of the 2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IEEE Computer Society, 2007, pp. 524–527.

[41] D. Cowan, M. Griss, B. Burg, Bluejade-a service for managing software agents, Tech. rep., Hp Inventors (2002).

[42] I. Nunes, C. Lucena, M. Luck, Bdi4jade: a bdi layer on top of jade, in: Proc. of the Workshop on Programming Multiagent Systems, 2011, pp. 88–103.

[43] S. Sguera, A. Stellato, D. Griesi, M. Pazienza, J-alinas: A jade-based architecture for linguistic agents, in: CEUR Workshop Proceedings, Vol. 204, CEUR, 2006, pp. 83–89.

[44] T. Marir, A. E. H. Silem, F. Mokhati, A. Gherbi, A. Bali, Norjade: An open source jade-based framework for programming normative multi-agent systems, International Journal of Open Source Software and Processes (IJOSSP) 10 (2) (2019) 1–20.

[45] P. Jedrzejowicz, I. Wierzbowska, Jade-based a-team environment, in: International Conference on Computational Science, Springer, 2006, pp. 719–726.

[46] G. Caire, D. Gotta, M. Banzi, Wade: a software platform to develop mission critical applications exploiting agents and workflows, in: Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems: industrial track, 2008, pp. 29–36.

[47] C. Guo, Z. Lin, K. Guo, Research and implementation of an enterprise-class mas application development framework–jadeee, in: International Conference on Computer Supported Cooperative Work in Design, Springer, 2007, pp. 294–303.

[48] L. S. Melo, R. F. Sampaio, R. P. S. Leão, G. C. Barroso, J. R. Bezerra, Python-based multi-agent platform for application on power grids, International Transactions on Electrical Energy Systems 29 (6) (2019) e12012.

[49] A. Gehani, D. Tariq, Spade: support for provenance auditing in distributed environments, in: ACM/IFIP/USENIX International Conference on Distributed Systems Platforms and Open Distributed Processing, Springer, 2012, pp. 101–120.