25 Years of Declarative Agent Technologies in Italy

Davide Ancona[1], Daniela Briola[2], Angelo Ferrando[3], Maurizio Martelli[1], and Viviana Mascardi[1](✉)

[1] Università degli Studi di Genova, Genova, Italy
{davide.ancona,maurizio.martelli,viviana.mascardi}@unige.it
[2] Università di Milano Bicocca, Milan, Italy
daniela.briola@unimib.it
[3] Università di Modena e Reggio Emilia, Modena, Italy
angelo.ferrando@unimore.it

Abstract. Since the first appearance of agents and multiagent systems (MAS), their close connection with declarative technologies was clearly devised and drove – and still drives – a large share of the research in the field. Declarative technologies play many roles in the MAS context: they can be exploited in the specification of the agents and MAS architecture, in the specification of their conversational flow, in their implementation, and in the representation of the agents' knowledge. In all of these roles, one main advantage of declarative technologies is the support to transparency and explainability thanks to a (more, w.r.t. other paradigms) readable, understandable, and interoperable description of the system, and to reliability, by means of formal verification of its properties. This chapter, guided by works presented at Workshop on Objects and Agents (WOA) starting from 2000, discusses the crucial role that declarative technologies play in a MAS context.

Keywords: Declarative approaches · Logic-based approaches · Cognitive and BDI Agents · Verification · Ontologies · Argumentation

1 Introduction

According to the widely accepted definition by Jennings, Sycara, and Wooldridge [113], an intelligent agent is a computer system situated within an environment, capable of exhibiting autonomous, social, reactive, and proactive behaviours. The strong definition of agents further incorporates the requirement of being modelled in terms of beliefs, desires, goals, and intentions, usually ascribed to humans. The Belief-Desire-Intention (BDI) model [158] is the most known architecture for cognitive agents.

AgentSpeak(L) [157] is one of the first agent oriented programming (AOP) languages for designing and implementing BDI agents and Jason [38] is one of the most well known interpreters for AgentSpeak(L). Jason has been integrated

V. Mascardi and A. Omicini (Eds.): *The Agents Journey*, LNCS 16395, pp. 287–319, 2026.
https://doi.org/10.1007/978-3-032-22940-3_11

with CArtAgO [162,163] for modelling and implementing the MAS environment as a set of artifacts, and with MOISE [111] to model the MAS organisational constraints, resulting into the JaCaMo framework [34].

Notably, the idea of integrating Jason, CArtAgO and MOISE appears for the first time in a WOA paper dating back 2010 [153], three years before the first paper on JaCaMo [35].

The study of intelligent agents, especially those built using the BDI model, provides a foundation for understanding how autonomous systems can reason and act within complex environments. These agent-based systems, particularly within the JaCaMo framework, emphasise transparency and modularity in their design, enabling explicit modelling of beliefs and intentions. However, as AI advances, especially with the rise of black-box models like deep neural networks, the focus has shifted towards balancing these foundational principles with the need for explainability in more opaque systems. This shift has sparked growing interest in combining traditional agent-based approaches with newer AI models to address challenges related to transparency and trust. These concerns become critical in high-stakes fields like healthcare, justice, and finance, where explainability is essential [106].

Declarative approaches in AI and MAS offer an advantage by focussing on *what* needs to be done rather than *how* to do it. This makes decisions more interpretable and rule-based. Unlike black-box models, declarative systems allow for explicit inspection and modification of decision-making processes, fostering greater transparency and trust. This approach aligns with the growing demand for Trustworthy AI (TAI), as outlined by the European Commission's Guidelines for Trustworthy AI[1]. These guidelines emphasise the importance of transparency, robustness, and fairness in AI systems.

Declarative models also promote fairness by making biases easier to identify and correct through explicit rule inspection, addressing a major shortcoming of black-box methods. Additionally, their modularity allows for simpler updates, enhancing reliability and robustness.

The Trustworthy AI framework emerged from both academic and industry efforts to address ethical concerns and establish principles for AI use across various domains [94,184]. Various approaches have been proposed to achieve TAI. For example, Vianello et al. [182] present a design framework based on user explainability and normative ethics, while Thiebes et al. [177] outline five foundational principles for TAI: beneficence, non-maleficence, autonomy, justice, and explicability. Similarly, Yeung [187] emphasises responsible stewardship by highlighting respect for human rights and democratic values.

Rawal et al. [159] introduced an eXplainable AI (XAI) framework that enhances trustworthiness by integrating principles such as transparency, accountability, and privacy. This framework bridges technical solutions with ethical concerns and incorporates legal standards, aligning AI systems with societal values.

Additionally, IBM's open-source toolkit for evaluating fairness and mitigating bias in machine learning models [25] exemplifies practical solutions. This effort

[1] https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai.

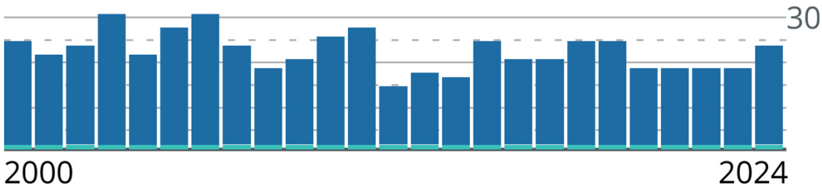

Fig. 1. Distribution of the 453 WOA papers in the range 2000–2024, from DBLP.

complements the European Commission's Guidelines for Trustworthy AI [96], which define seven key requirements for AI systems to be deemed trustworthy.

The chapter is organised as follows. Section 2 presents the WOA papers we took under consideration in this chapter and analyses them from a quantitative point of view, whereas Sect. 3 provides a qualitative evaluation of these works. Section 4 discusses our own research, always using WOA publications as fil rouge. Finally, Sect. 5 outlines the future directions of the research on declarative technologies and MAS.

2 A Look into Raw Data

We start our journey with a look into raw data. We analysed the titles and abstracts of all the papers published at WOA from 2000 to 2024, looking for relevant keywords and their syntactic variants. The results of our search are more precise for papers from 2005 onwards, as these papers' data were automatically collected into a single spreadsheet from Scopus, and this made the analysis easier. For papers published in the first five editions, data collection was manual and hence more error prone. The extensive results are shown in Table 1, Table 2, Table 3 in the Appendix: a summary is provided in the sequel.

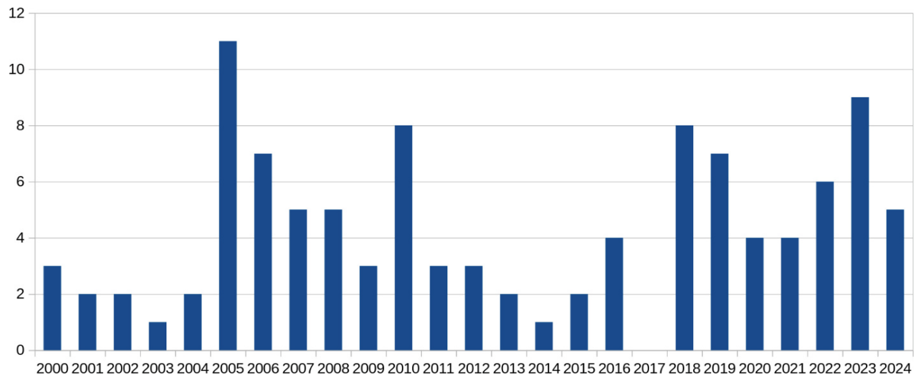

Fig. 2. Distribution of the 107 WOA papers dealing with declarative agent technologies in the range 2000–2024.

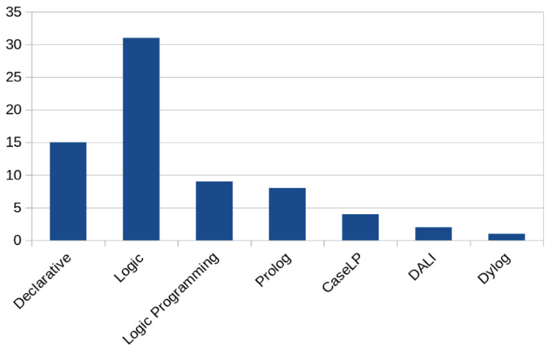

Fig. 3. WOA papers dealing with declarative and logic-based approaches. The distinct papers belonging to this category are 54.

Out of 453 WOA papers indexed in DBLP,[2] whose distribution over years is shown in Fig. 1, 107 included at least one of the keywords matching our search in their title or abstract. Their distribution over years is shown in Fig. 2.

The five areas we considered to look for papers and to group them into coherent categories, are **declarative and logic-based agents**, **cognitive and BDI agents**, **verification**, **ontologies**, and **argumentation**. These areas are not disjoint, and many papers belong to more than one.

Figure 3 shows the number of papers that include ‘declarative’, ‘logic’, or ‘logic programming’ in their title or abstract. Papers that deal with logic in general, but not with logic programming, are counted only in the ‘logic’ sub-category, and vice-versa. Papers that deal with both, contribute to increasing both counters. Sub-categories are not disjoint, and they show that the ‘logic’ (meant in a general, broad sense) keyword is the most popular one, appearing in 31 papers, followed by ‘declarative’ (15 papers) and ‘logic programming’ (9). ‘Prolog’ appears in 8 papers, while – notably – no paper mentions ‘answer set’ or ‘stable model’ at least in its title or abstract. We read the abstracts and titles of these papers, and a few programming languages and tools designed and implemented by members of the WOA community appeared almost often: DCaseLP [3,18,44,129], DALI [36,83], DyLOG [172]. They are included in Fig. 3 for reference.

Figure 4 presents figures related with papers that include ‘BDI’ (or the extended version of the acronym) or ‘cognitive’ in their title or abstract. By reading these abstracts, we realised that many papers mentioning AgentSpeak(L), Jason, or JaCaMo, did not result from our search because they did not also include the ‘BDI’ keyword. We searched for these three languages/technologies explicitly and we increased the BDI count any time one paper mentioned them: indeed, one paper where AgentSpeak(L) is used, is one paper on BDI agents. In Fig. 4, PRACTIONIST also appears because it was mentioned in 4 regular papers [127,140–142].

[2] https://dblp.org/db/conf/woa/index.html.

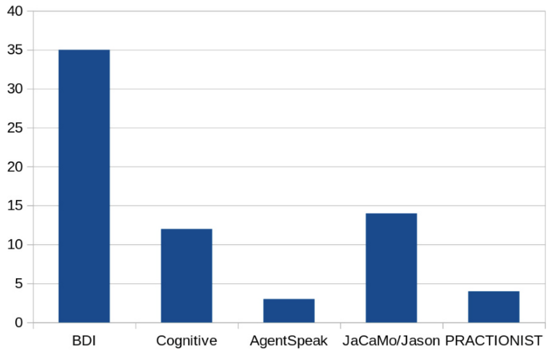

Fig. 4. WOA papers dealing with cognitive and BDI agents. The distinct papers belonging to this category are 42.

Only 13 distinct papers out of 107 deal with formal verification. The keyword recurring most often is 'verification' in general (11 papers), followed by 'model checking' and 'conformance' (3 papers both). Figure 5 shows the histogram of these data.

The distinct papers dealing with semantic web and ontologies are 29: in 26 titles/abstracts the words 'ontology' or 'OWL' appear, in 8 of them the 'semantic web' keyword appears (see Figure Fig. 6). Of course, some papers mention both. To make our search as exhaustive as possible we also looked for 'description logic' but it is not mentioned, at least in titles and abstracts.

We considered the argumentation research area, given that argumentation frameworks are often, although not always, based on logic. Only 3 papers belong to this category, which we did not divide into sub-categories.

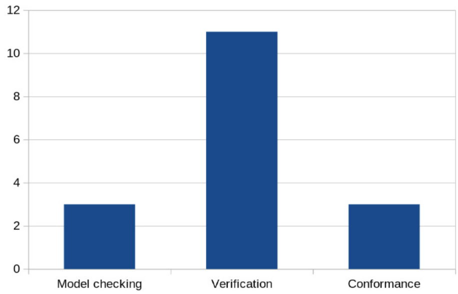

Fig. 5. WOA papers dealing with formal verification. The distinct papers belonging to this category are 13.

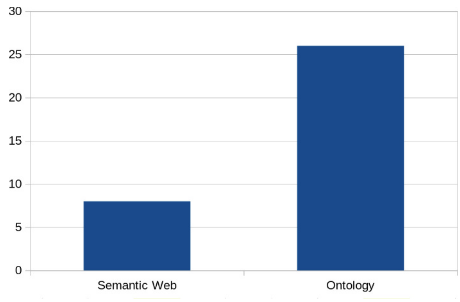

Fig. 6. WOA papers dealing with semantic web and ontologies. The distinct papers belonging to this category are 29.

Figure 7 shows that out of the 107 papers we considered in this work, 3 deal with topics in the intersection of 3 different categories, 28 cover two distinct categories, and the other 76 papers belong to one category only. Out of the 3 papers covering 3 categories, 2 deal with declarative and logic-based agents ∩ cognitive and BDI agents ∩ ontologies [112, 121], while one deals with declarative and logic-based agents ∩ cognitive and BDI agents ∩ verification [18].
The distribution of the 28 papers falling in two categories is

11 papers in declarative and logic-based agents ∩ cognitive and BDI agents;
6 papers in cognitive and BDI agents ∩ ontologies;
5 papers in declarative and logic-based agents ∩ verification;
3 papers in declarative and logic-based agents ∩ ontologies;
2 papers in declarative and logic-based agents ∩ argumentation;
1 paper in cognitive and BDI agents ∩ verification.

Finally, most papers belonging to one category only fall either in the declarative and logic-based agents one (30 papers), in the cognitive and BDI agents one (21 papers), or in the ontologies one (18 papers).

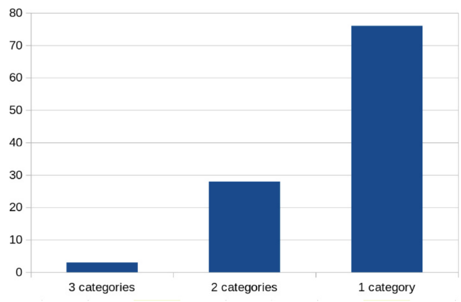

Fig. 7. Number of papers belonging to exactly one, two, or three categories among declarative and logic-based agents; cognitive and BDI agents; verification; ontologies; argumentation.

We conclude our journey by mentioning the research impact of these papers in terms of citations. Due to lack of indexing of the papers belonging to the first five editions, and to the recent publication of the 2024 proceedings, we only considered the editions from 2005 to 2023 included, for 92 papers in total. According to Scopus, at the time of writing these 92 papers obtained 5.4 citations per paper on average, with one paper obtaining more than 110 citations and 3 papers obtaining between 30 and 50 citations. The average number of citations of papers published on the International Conference on Autonomous Agents and Multiagent Systems (AAMAS) Proceedings is 11.2. With due proportions between WOA and an A* ranked conference as AAMAS, the impact of WOA publications on declarative approaches to agent technologies and languages seems extremely significant.

3 From Data to Knowledge

In this section we organise raw data in a timeline, finding semantic connections among them and positioning the WOA contribution in the wider international panorama. Papers that are not dealt with in details in this book are discussed more in depth, and cross-references to other chapters are inserted when relevant.

3.1 Declarative and Logic-Based Agents

The importance of declarative and logic-based agent languages and technologies was recognised as early as the notion of intelligent agent was conceived. It was already clear in the seminal paper by Wooldridge and Jennings [185], that – referring to agent architectures – wrote

> *The close relationship between symbolic processing systems and mathematical logic means that the semantics of such architectures can often be represented as a logical system of some kind.*

Along this line, in the same year Kowalski presented the logical foundations of MAS [116] and, one year later, he addressed the problem of how logic programming and meta-programming in particular could be adopted to reconcile reactivity and proactivity (rationality) in agents [117,118].

The Italian research community has been traditionally very active in declarative approaches and in logic programming, with the "Gruppo Ricercatori e Utenti di Logic Programming" association dating back 1986. When the notion of intelligent agent was put forward, many Italian scientists working on declarative approaches started to apply them to the specification, prototyping, implementation and verification of agents and MAS. Most of them, are still doing.

It is not a case that the first edition of the Declarative Agent Languages and Technologies (DALT) workshop was organised in 2003 by scientists with a strong background in logic programming and Prolog, and that two of them were Italian. Out of ten DALT editions, nine were co-organised by Italian scientists. Today, the Preface of the first DALT edition is still (maybe, even more) valid in its statement *the growing complexity of agent systems calls for models and technologies that promote system predictability and enable feature discovery and verification. Formal methods and declarative technologies have recently attracted a growing interest as a means to address such issues.*

The active contribution of the Italian community to the development of the research area is also witnessed by recent surveys [54].

WOA mirrors, at the local scale, the evolution of the field at the international scale.

Tools and Languages. Among the oldest WOA papers in this category, most proposed either languages for specifying and programming agents [3,83,129,172], or languages for specifying and verifying agent interaction protocols [18,67,107], or architectures [10] and applications [139,167]. Among agent programming languages, the already mentioned DALI is discussed in detail in [183], while CaseLP will be discussed later in this chapter.

Coordination. Coordination is also a relevant problem to address, and is achieved via abductive reasoning [68] or via the TuCSoN [147] infrastructure [143].

Neuro-Symbolic Integration and Other Recent Trends. In the most recent papers, besides languages [55,70], platforms and architectures [53,120,121,137], applications [87,156] that are still relevant for the community, we observe an increasing number of proposals that cross-breed neural and symbolic approaches. The importance of this topic is also witnessed by [1], which is the chapter of this book devoted to it.

This integration is often aimed at making sub-symbolic AI explainable: an example of integration of symbolic and sub-symbolic techniques in the XAI context is presented by Calegari et al. [50] while Pisano et al. [152] present a set of guidelines based on logic induction and logic constraints aimed at building explainable intelligent systems even when they exploit sub-symbolic techniques, and discuss a working prototype. PSyKE [166] supports extraction of symbolic knowledge in logic form from different sorts of black-box predictors. Finally, an analysis of problems from the field of computational logic that may benefit from graph-related problems where Graph Neural Networks (GNN) has been proved effective, along with the application of GNN to logic theories via an end-to-end toy example, are presented by Agiollo et al. [2]. Fewer works target Reinforcement Learning, either for capturing a recursive pattern in neural-symbolic RL [27] or for integrating a runtime verification tool with OpenAI Gym, a popular framework for developing and comparing RL algorithms [180].

Works addressing games [65], the law application domain [56,86], the exploitation of Theory of Mind to boost collaboration [108], are also worth mentioning among the most recent contributions to WOA, belonging to the 'declarative and logic based agents' category.

3.2 Cognitive and BDI Agents

Due to the intersection among categories that we discussed in Section Sect. 2, some papers dealing with cognitive and BDI agents also deal with verification and with knowledge representation, and will be discussed later. In this section we overview those works with no overlaps with the categories discussed in Sections Sect. 3.3 and Sect. 3.4.

The history of BDI agents and languages is covered by many surveys. Besides an old one presented at WOA in 2005 [134], recent works involving Italian scientists [37,62] represent good readings for addressing the topic.

Tools and Languages. Among the works presented at early WOA editions we mention PRACTIONIST [127,140–142]. PRACTIONIST stands for PRACTIcal reasONIng sySTem; it is an old system supporting the development of BDI agents in Java (using JADE [26]) with a Prolog belief base. A goal-oriented approach drives the PRACTIONIST reasoning cycle, that clearly separates the deliberation and the means-ends reasoning, as well as the states of affairs to pursue and the way to do it. Works involving the Jason and CArtAgO (JaCa) environments, and their integration with web services, are also relevant examples of papers in this category [138,154]. A security mechanism for SOAP-style and REST-style web services that allows the distribution of the delegation of access rights among different rational agents is presented by Tomaiuolo and Turci [179].

BDI Agents and Robots. A more recent strand of research involves the theoretical and practical relations between BDI agents and robots. On the practical side, Alzetta and Giorgini propose an integraton of the BDI model into one of the most popular robotics frameworks, ROS 2 [5], while works by Falcone and colleagues [59,64] illustrate how to include self-modelling skills involved in trustworthy interactions, and how to exploit principles underlying theory of delegation, theory of mind and BDI agent modelling, to better shape the interactions with the humanoid robot Nao. The theoretical work by Cantucci and Falcone [61], on the other hand, outlines the development of computational cognitive models aimed at supporting a useful, effective, acceptable and trustworthy interaction between humans and robots.

Applications. Applications are always in the agenda, and range from cultural heritage domain [6,60] to interaction in natural language [122,173].

3.3 Formal Verification

Formal verification is the branch of formal methods focussed on verifying whether a system – or a model of the system – meets its specifications. Verification can be exhaustive, as in model checking or theorem proving, where the entire model is explored to ensure the property holds as specified. This means that all possible execution paths or states are considered, though the property itself may require verification on some paths (existential) or all paths (universal), depending on the formula. In contrast, non-exhaustive approaches, such as runtime verification or software testing, examine only a subset of possible executions – often limited to specific test cases or the current execution in runtime verification.

The importance of formal verification of agents and MAS has been discussed in many works [115], also in connection with the pressing need of certifying autonomous software systems [103]. As expected, many papers presented at WOA address this topic.

Model Checking. One of the oldest works published at WOA in this category deals with the application of model checking to the cryptographic mechanisms designed for the protection of mobile agents from their environment, with a particular emphasis on agent data integrity [125]. In details, the authors check the properties of a protocol model proposed by Corradi et al. [72] using a model checker developed at Polytechnic of Torino.

Probabilistic model checking is exploited by Casadei and Viroli [63]. The specification language used to model computational fields in pervasive and spatial systems is a variant of the PRISM language [110]; a generation process translates the model into a PRISM module and then builds a complete PRISM specification representing the whole network, ready to be model checked.

VITAMIN (VerIficaTion of A MultI-ageNt system [100]) is a formal verification framework integrating model checking algorithms for many different logics and model formalisms while providing a user-friendly experience. In that paper, VITAMIN is exploited in the robotics domain, focussing on an extension of Alternating-time Temporal Logic with resource bounds (RB-ATL).

Protocol Conformance. A bunch of papers addressed the problem of verifying a priori conformance of *atomic services* offered by individual agents and represented as finite state automata against a global interaction protocol (a choreography), also represented as a finite state automaton [17,18]. Given such a representation, a priori conformance test amounts to answering in a positive way to the following questions:

- Is it possible to verify that a service, playing a role in a given global protocol, produces at least those conversations which guarantee interoperability with other conformant service?
- Will such a service always follow one of these conversations when interacting with the other parties in the context of the protocol?
- Will it always be able to conclude the legal conversations it is involved in?

An extension of a priori protocol conformance that takes into account the possibility for agent names and messages to have different syntax, but equivalent (hence the 'modulo mapping') semantics, is presented in [8]. Conformance check is fully implemented in SWI-Prolog.

Linear Time Logic is used in [15], where – in the domain of declarative presentation of curricula – model checking techniques are used to verify that the user's learning goal is supplied by a curriculum, that a curriculum is compliant to a curricula model, and that competence gaps are avoided.

Protocol Combination. In [107] interaction protocols are defined by a set of temporal constraints, which specify the effects and preconditions of the communicative actions on the social state. The problem of combining protocols to define new more specialised protocols is addressed and the approach is applied to the specification and verification of clinical guidelines. The work by Chesani et al. [67] is similar in the adoption of a logic-based framework and also in the medical guidelines application domain.

Runtime Verification. Papers on runtime verification appeared only recently at WOA, coherently with the recent take-up of this approach w.r.t. more consolidated techniques. In [7], we address the problem of decentralised runtime verification of agent interaction protocols by means of MAS-DRiVe, an algorithm implemented in SWI-Prolog that partitions a MAS into sub-MASs which can be monitored independently. The already cited RMLGym [180] allows users to define reward functions using RML specifications [9] and then generates reward monitors that evaluate the agent's performance and provide feedback at each step.

Testing and Validation via Explanations. Although not based on formal methods, we mention the conceptual framework, supported by a working prototype, proposed by Yan et al. [186]. The tool creates narratives explaining the behaviour of a Jason agent at multiple levels of abstractions, meant to be useful at different levels – developers, designers, users. During the testing and validation phases, developers can use the explanation of the agents, take the narrative, and check the requirements to see if they meet them, or compare the narrative with the user stories of the system to ensure alignment. While this must be made manually, so far, there is room for taking advantage of past experience of conformance check and formal verification, to make the above tasks more automatic and well grounded.

3.4 Semantic Web and Ontologies

The close relation between agents and semantic web technologies, especially ontologies, is as old as the semantic web itself [109], and is witnessed by many publications since then [85,135].

FIPA and Ontology Services. In 2000, the first FIPA Ontology Service Specification was released[3]. In the early WOA editions, many papers proposed FIPA-compliant architectures integrating ontology service providers [40,148,181] or frameworks that – albeit not being FIPA-compliant – aimed at offering services for retrieving, managing, mapping ontologies in MAS with mobile agents [74] or in service-oriented architectures [155,178]. More recently, Ruta et al. [164] follow the service-oriented architecture research line by proposing a framework where device agents self-organise in social relationships, interact autonomously and

[3] http://www.fipa.org/specs/fipa00086/XC00086C.pdf.

share information, cooperating and orchestrating ambient resources. A a service-oriented architecture allows collaborative dissemination, discovery and composition of service/resource descriptions by device agents. Decision and choreography capabilities of software agents leverage Semantic Web languages at the knowledge representation layer.

Ontologies and Agent Languages. JADEL [30] is the ancestor of the Jadescript language also presented at WOA [33,149], and not only [29,32]. JADEL and JadeScript provide ontologies not only to enable fruitful communication among agents, following the FIPA specifcations, but also to define structured data that can be used in the construction of the agents and their behaviours. In order to support this extended approach to the use of ontologies, Jadescript treats ontologies as organised packages of concepts, actions, predicates, and propositions.

Moving from JADE to the Jason (and its ecosystem) language, Chella et al. [66] implement a controlled semantic system to manage the belief base of a MAS at runtime. Their goal is achieved by interfacing Jason, CArtAgO and the Jena library[4] for managing OWL ontologies.

Ontology Models. In [79,161] the Problem Ontology metamodel, inspired by the FIPA standard, is used for describing the problem domain by a set of meta classes (Concept, Action and Predicate) in a BDI context. OntologyBeanGenerator 5.0 [43] is also based on the model driven approach – where the model is represented by an ontology – and injects programming components inside an ontology: OntologyBeanGenerator 5.0 extends the OntologyBeanGenerator plugin for Protégé in order to generate a Java representation of an OWL ontology for Jade, and supports the modelling of exceptions, formalised at the ontology level, and of methods associated with ontology elements, to set the interface of artifacts at design time.

Applications and Other Recent Trends. Applications where ontologies are exploited in combination with agents and MAS span very diverse domains. In [126] an ontology based on fuzzy logic is used inside a tool that analyses different crowded scenarios – in particular, concerts – according to crowd features like density and duration.

In [124], automatic interpretation and execution of laboratory protocols expressed in XPDL+OWL in the life sciences domain is carried out thanks to a MAS implemented with JADE and WADE [49].

Loseto et al. designed and implemented an agent able to perform an automated profile annotation by adopting Semantic Web languages, and tested it in a domotic environment [123].

PRESTO [47] applies agent technologies to serious gaming by creating game independent, modular Non-Player Characters behaviours based on a BDI approach. Both the environment and the internal state of the NPC are modelled via ontologies [46]. VideOWL [131] classifies video games starting from their core features, hence belonging to a domain similar to PRESTO's one.

[4] https://jena.apache.org/.

The Indiana MAS project [132] and the VEsNA framework [112], discussed in Section Sect. 4, also integrate ontologies and semantic web technologies.

3.5 Argumentation

The three WOA papers that – based on their title and abstract – deal with argumentation are all co-authored by Omicini and cover a range of 14 years, from 2008 to 2021.

In the oldest one [144] the authors present a system for Alternative Dispute Resolution exploiting a Co-Argumentation Artifact and a Dialogue Artifact to coordinate the agents during the argumentative process. The technological support for the artifacts is provided by the already mentioned TuCSoN infrastructure. For each argument set, the argument validity; the relations of undercut and attack between arguments; the conflict-free sets; and the preferred extensions are computed in Prolog, via metaprogramming.

Pisano et al. [150] address the problem of cooperative argumentation in the context of a MAS. The logic-based agreement framework Arg2P [151] evaluates the admissibility of a single statement without building the entire argumentation graph: starting from the Arg2P single query evaluation mode, they introduce the corresponding distributed computational model. In their WOA paper, they show how the argument evaluation algorithm of Arg2P can be parallelised, and present a complete model for decentralised reasoning based on the actor model.

Finally, the work by Calegari et al. [56] tackles argumentation from a theoretical point of view: it introduces the concept of computable law as an argumentation-based MAS and investigates the most promising approaches and technologies – among which, knowledge representation and ontologies, machine learning, argumentation models and technologies, human-computer interfaces – that may be exploited for engineering of systems and services.

4 From CaseLP to VEsNA

Nomen omen. The story of our research group starts with the design and development of CaseLP,[5] whose name has a double reading. On the one hand, the acronym expands into 'Complex Application Specification Environment based on Logic Programming', stressing our interest in developing applications that are complex enough to be challenging and exciting, and in exploiting Logic Programming for achieving our goal. On the other hand, 'Case' intentionally reminds to 'Computer Aided Software Engineering', emphasising the engineering dimension of our research, that – over years – covered the theory and practice of MAS specification, implementation, and verification, often in collaboration with industrial partners.

The first paper mentioning CaseLP was published in 1998 [130]. From a descriptive point of view, a CaseLP agent is characterised by a *kind* (logical,

[5] https://person.dibris.unige.it/mascardi-viviana/Software/DCaseLP.html.

interface, facilitator or manager), an *architecture* (data structures that form the agent's internal components + control flow), an *interpreter* and a *set of services*. From a computational point of view it consists of a *state*, a *behaviour* and an *engine*, that are all architecture-dependent components. Private and public *mail-boxes* are further components, needed to ensure asynchronous communication among agents. CaseLP comes with a simple engineering methodology, whose steps include: 1. identification of the set of agents and their interconnecting structure; 2. choice of the communication protocol among each pair of communicating agents; 3. specification of the behaviour of each agent in the system; 4. implementation of the prototype; 5. execution of the obtained prototype.

After 26 years, our core business is still engineering complex applications modelled and implemented as MAS, by exploiting symbolic approaches when it makes sense (and other approaches when they are more suitable). In the meantime, the world evolved, and we evolved too.

In this section, we illustrate our evolution following the most significant papers that we presented at WOA from 2000 to 2024.

Our starting point is HEMASL [129], a meta-language for specifying agent architectures in an imperative way, and translating them into a set of clauses in $\mathcal{E}_{hhf}$ [91], a language based on linear logic. $\mathcal{E}_{hhf}$ specifications can be tested and/or verified and refined in CaseLP and finally translated into Prolog, for being executed. The distributed version of CaseLP, DCaseLP, was first presented at WOA 2002 [3]. DCaseLP integrates both imperative (object-oriented) and declarative (rule-based and logic-based) languages, as well as the UML graphical notation.[6] The rationale behind DCaseLP is that MAS development requires engineering support for a diverse range of non-functional properties, such as understandability of the MAS at various conceptual levels, integrability of heterogeneous agent architectures, usability, re-usability, and testability. By providing the MAS developer with a set of languages, and allowing for the choice of the most suitable language to model, implement, or test each property, DCaseLP heads towards a modular approach to Agent-Oriented Software Engineering [114].

The development of approaches and tools to verify the agents' behaviour – whose importance had already been devised in the early papers on HEMASL and CaseLP – characterised a long and fruitful collaboration with researchers from Torino [18] and from other universities [13].

While working on the more theoretical aspects of MAS specification and verification, we also managed to design and implement real MAS, in collaboration with industries or within funded projects. Two of these projects were developed together with Ansaldo STS S.p.A., now Hitachi Rail STS S.p.A.. The first started in February 2008 and ended in September 2008, and consisted in a MAS prototype implemented in DCaseLP that monitors processes running in a railway signalling plant, detects functioning anomalies, and provides support to the early notification of problems to the Command and Control System Assistance [44]. The second was a multiagent resource allocation problem [42], solved

[6] https://www.uml.org/.

by means of a complex negotiation protocol named FYPA (Find Your Path, Agents!) among the involved entities. The problem is described as "*For each entity that enters the graph from a start point either confirm the validity of the plan stated in the static allocation plan, or, if some unexpected event occurred that makes the original plan no longer applicable, find a new plan for reaching an end point of the graph. The new plan should minimise the delay in which the entity exits the graph and the number of required changes with respect to the original plan, as well as the number of entities involved in the reallocation process.*" [45]

Due to an NDA, we could not be explicit on what entities and graph edges represent. They can be easily guessed, given that Ansaldo STS dealt with trains that enter rail stations on pre-allocated tracks, which may no longer be available when the train is approaching, hence triggering a re-planning. A NetLogo interface for a standalone version of FYPA was also developed [41].

A project developed with SELEX Elsag S.p.A., now Leonardo S.p.A., consisted in the Jade implementation of a MAS for solving unavailability problems in a scenario inspired by the electricity power network domain [119].

The Indiana MAS project obtained a funding of 747.600 euros by the Italian Ministry of Education, University and Research "Futuro in Ricerca 2010" program.[7] It provides a framework for the digital protection and conservation of rock art natural and cultural heritage sites, by storing, organising and suitably presenting information about them. Its objectives, illustrated at WOA 2012 [132], were to develop a MAS supporting archaeologists and historians in 1. integrating heterogeneous unstructured data (multilingual textual documents, pictures, and drawings) related to rock carvings into a single repository; 2. normalising data by recognising those referring to the same object, correctly associating them with its digital representation, and removing duplicate data; 3. classifying normalised data according to the "Indiana ontology"; 4. organising classified data into a Digital Library and making the library accessible thanks to a web-based, multilingual, user-friendly interface; 5. interpreting data stored in the Digital Library, finding relations among them, and enriching them with the semantic information extracted thanks to this interpretation and relation retrieval stage. The project successfully achieved its goals: videos showing demos be found in the project's web site,[8] together with all the project's papers and deliverables.

We attended WOA on a regular basis in these 25 years: many papers published after 2012 dealt with verification and ontologies, and have been already discussed in Section Sect. 3.

Our most recent contribution, VEsNA [99,105,112], can be seen as a CaseLP 4.0: it is still conceived as an environment for engineering software prototypes of complex applications, and is still (also) rooted on logic programming but – as for Industry 4.0 – it pays more attention to human-computer interaction, to the continuum between physical and digital worlds, and to safety.

7 http://indianamas.dibris.unige.it/.
8 http://indianamas.dibris.unige.it/index.php/the-indianamas-demonstrator.

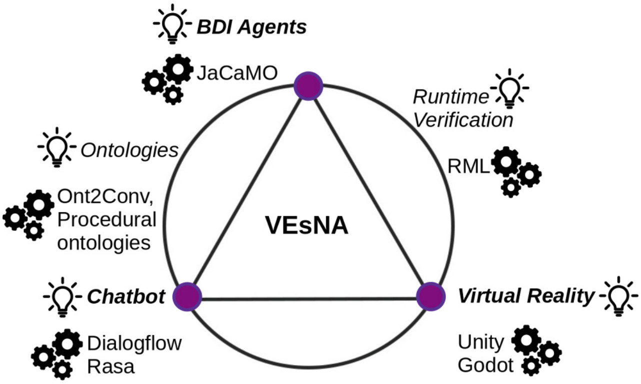

Fig. 8. VEsNA concept.

VEsNA is a general-purpose and agent-based framework for managing Virtual Environments via Natural language Agents. As shown in Fig. 8, VEsNA builds upon three pillars:

(i) a chatbot-like interface to support the sociality of agents; thanks to the chatbot supporting natural language interaction, sociality is not limited to agent-agent conversations, but it is extended to agent-human ones;
(ii) a framework for building the dynamic virtual environment; virtual reality is very suitable for supporting agents' situatendness and embodiment: therein, agents are expected to perceive the reality and to act over it;
(iii) a framework for implementing cognitive, declarative agents able to reason about knowledge (about what users say, about the environment, about themselves) and to provide human-readable explanations.

The chatbot interface can be implemented in both Dialogflow[9] and Rasa[10], and the virtual environment in both Unity[11] and Godot[12]. The framework for BDI agents is JaCaMo.

A mechanism for verifying the behaviour of the system's components, at runtime is also integrated in the VEsNA concept: runtime verification is supported by the adoption of the Runtime Monitoring Language, RML [9], compiled down into SWI-Prolog and whose monitoring engine is implemented in SWI-Prolog too.

The presence of ontologies in VEsNA was considered since the very beginning as a tool to provide a unique specification of the conversation domain, and generate skeletons for both the Jason agents in the backend and the chatbot frontend from it [95]. In our WOA 2024 paper, we illustrated the possibility to use ontologies also to describe agents' behaviours [112].

[9] https://cloud.google.com/dialogflow.
[10] https://rasa.com/.
[11] https://unity.com/.
[12] https://godotengine.org/.

All the VEsNA components are implemented, working, and available online under GPL 2.0 license, although they are not all fully integrated into a unique repository, yet[13].

The technological panorama changed dramatically from 2000 to nowadays, with the perceived and narrated supremacy of sub-symbolic approaches over symbolic ones, and the Large Language Models breakthrough in the last few years. We evolved and did our best to adapt to these changes, but we never retreated from our original vision of science and of agent oriented software engineering. We are confident we will continue to keep the pace: technology goes fast, but also VEsNA does!

5 Future Perspectives and Conclusions

Declarative approaches have proven to be essential in fields like MAS due to their emphasis on transparency, interpretability, and rule-based decision-making. Unlike black-box models, declarative systems allow for explicit reasoning about the system's behaviour, enabling easier identification and correction of biases and hence supporting trust and accountability. These characteristics are increasingly vital as AI systems are deployed in high-stakes domains where explainability and reliability are crucial.

In the MAS area, declarative models have been particularly successful in providing modularity and flexibility, enabling systems to adapt more easily to new rules and environments. Their ability to model complex organisational constraints, as demonstrated by frameworks like JaCaMo, highlights the practical advantages of declarative approaches in managing distributed, autonomous agents.

Looking to the future, one potential direction lies in neuro-symbolic AI, where declarative and logic approaches could complement the data-driven nature of machine learning, providing structured, rule-based reasoning alongside the adaptability of neural networks. This integration could address the opaque nature of black-box models, making AI systems more interpretable while preserving their powerful pattern recognition capabilities.

Another exciting direction is the application of declarative approaches in the development of chatbots and conversational agents. These systems, widely used today, often struggle with issues like unpredictability, lack of coherence, and ethical concerns. By combining the flexibility of machine learning-driven chatbots with the robustness, inspectability and explainability of declarative methods, we can create conversational agents that are not only more reliable but also capable of motivating their decisions and maintaining consistency with human values.

[13] https://github.com/VEsNA-ToolKit.

In summary, declarative approaches offer powerful tools for building trustworthy and reliable AI systems. As AI continues to evolve, their integration with other advanced techniques, particularly in areas like machine learning and conversational systems, is likely to unlock new possibilities for creating more responsible and effective intelligent systems.

Acknowledgments. This work was partially supported by the following projects:

– *T-LADIES – Typeful Language Adaptation for Dynamic, Interacting and Evolving Systems*, funded by the Italian MUR program PRIN 2020 under grant number 2020TL3X8X

– *ENGINES – ENGineering INtElligent Systems around intelligent agent technologies*, funded by the European Union – Next Generation EU within the framework of the National Recovery and Resilience Plan NRRP – Mission 4 "Education and Research" – Component 2 – Investment 1.1 "National Research Program and Projects of Significant National Interest Fund (PRIN)" – Call PRIN 2022 – D.D. n. 104 of 02/02/2022, under grant number 20229ZXBZM

– *FAIR – Future Artificial Intelligence Research*, PNRR MUR Project PE0000013 funded by the European Union – NextGenerationEU, CUP J33C24000420007

Appendix

In this Appendix we list the 107 papers published at WOA from 2000 to 2024, dealing with declarative agent technologies. Each column stands for one feature that may or may not characterise the paper (the ✓sign), based on our keyword-based search on the papers' titles and abstracts. Besides the reference and the publication year, tables show the family name that appears first in the list of authors as in most cases it helps identifying the research group that worked on the paper. The keywords we looked for, along with their syntactic and semantic variants, are:

Dec. Declarative;
Log. Logic (in general);
LP Logic Programming;
Prol. Prolog;
BDI;

Cog. Cognitive;
MC Model checking;
Ver. Verification;
Con. Conformance checking;
Ont. Ontology/OWL/Semantic Web;
Arg. Argumentation.

Table 1. Table 1

Ref.	1st Auth.	Yr	Dec.	Log.	LP	Prol.	BDI	Cog.	MC	Ver.	Con.	Ont.	Arg.
[10]	Armando	2000		✓									
[68]	Ciampolini	2000		✓									
[129]	Marini	2000		✓									
[139]	Montaldo	2001	✓										
[160]	Repetto	2001	✓										
[3]	Albertoni	2002				✓							
[125]	Maggi	2002								✓			
[83]	Costantini	2003			✓								
[172]	Schifanella	2004				✓						✓	
[178]	Tomaiuolo	2004										✓	
[18]	Baldoni	2005	✓	✓			✓			✓	✓		
[21]	Baldoni	2005		✓								✓	
[39]	Bouquet	2005		✓									
[48]	Cabitza	2005	✓										
[67]	Chesani	2005		✓						✓			
[71]	Cordì	2005										✓	
[74]	Corradini	2005										✓	
[84]	Costantini	2005		✓									
[107]	Giordano	2005		✓						✓			

Table 2. Table 2

Ref.	1st Auth.	Yr	Dec.	Log.	LP	Prol.	BDI	Cog.	MC	Ver.	Con.	Ont.	Arg.
[134]	Mascardi	2005		✓			✓						
[140]	Morreale	2005				✓	✓						
[17]	Baldoni	2006								✓	✓		
[81]	Costantini	2006	✓	✓									
[141]	Morreale	2006					✓						
[142]	Morreale	2006					✓						
[143]	Oliva	2006		✓			✓						
[175]	Sguera	2006										✓	
[181]	Vecchiola	2006										✓	
[15]	Baldoni	2007	✓	✓					✓	✓			
[57]	Cannata	2007		✓								✓	
[92]	Deufemia	2007		✓			✓						
[133]	Mascardi	2007										✓	
[155]	Poggi	2007										✓	
[19]	Baldoni	2008	✓										
[40]	Briola	2008										✓	
[44]	Briola	2008				✓							
[127]	Marguglio	2008					✓						
[144]	Oliva	2008				✓						✓	✓
[126]	Magnolo	2009										✓	
[148]	Passadore	2009										✓	
[154]	Piunti	2009						✓					
[13]	Baldoni	2010								✓			
[97]	Falcone	2010		✓				✓					
[102]	Fichera	2010	✓										
[124]	Maccagnan	2010										✓	
[138]	Minotti	2010					✓						
[153]	Piunti	2010					✓						
[168]	Santi	2010					✓						
[179]	Tomaiuolo	2010					✓						
[6]	Amato	2011					✓						
[16]	Baldoni	2011	✓										
[76]	Cossentino	2011					✓						
[12]	Baldoni	2012	✓										
[63]	Casadei	2012							✓	✓			
[132]	Mascardi	2012										✓	
[123]	Loseto	2013										✓	
[161]	Ribino	2013					✓					✓	
[79]	Cossentino	2014					✓					✓	
[46]	Busetta	2015					✓	✓				✓	
[171]	Sapienza	2015						✓					
[7]	Ancona	2016								✓			
[30]	Bergenti	2016										✓	
[55]	Calegari	2016			✓	✓							

Table 3. Table 3

Ref.	1st Auth.	Yr	Dec.	Log.	LP	Prol.	BDI	Cog.	MC	Ver.	Con.	Ont.	Arg.
[128]	Mariani	2016					✓						
[8]	Ancona	2018									✓		
[43]	Briola	2018					✓					✓	
[53]	Calegari	2018			✓								
[66]	Chella	2018					✓					✓	
[69]	Ciatto	2018			✓								
[77]	Cossentino	2018					✓					✓	
[122]	Longo	2018	✓				✓						
[164]	Ruta	2018										✓	
[5]	Alzetta	2019					✓						
[50]	Calegari	2019			✓								
[58]	Cantone	2019										✓	
[59]	Cantucci	2019					✓	✓					
[64]	Castelfranchi	2019					✓						
[78]	Cossentino	2019					✓						
[87]	D'Asaro	2019		✓									
[56]	Calegari	2020		✓								✓	✓
[70]	Ciatto	2020			✓	✓							
[120]	Longo	2020		✓			✓	✓					
[152]	Pisano	2020		✓									
[2]	Agiollo	2021			✓								
[121]	Longo	2021		✓				✓				✓	
[150]	Pisano	2021										✓	✓
[166]	Sabbatini	2021			✓								
[24]	Bella	2022										✓	
[60]	Cantucci	2022					✓	✓					
[65]	Catta	2022		✓									
[156]	Rafanelli	2022		✓									
[173]	Seidita	2022					✓	✓					
[176]	Sterling	2022		✓									
[27]	Beretta	2023	✓	✓									
[61]	Cantucci	2023						✓					
[82]	Costantini	2023		✓									
[131]	Martino	2023										✓	
[137]	Messina	2023	✓	✓									
[165]	Sabbatini	2023						✓					
[174]	Seidita	2023					✓						
[180]	Unniyankal	2023	✓	✓						✓			
[186]	Yan	2023					✓			✓			
[36]	Bordini	2024			✓	✓	✓						
[86]	Cristani	2024		✓			✓						
[100]	Ferrando	2024		✓					✓				
[108]	Grimaldi	2024		✓			✓						
[112]	Ivan	2024	✓					✓				✓	

References

1. Agiollo, A., Calegari, R., Ciatto, G., Magnini, M., Omicini, A., Sabbatini, F.: Intelligent agents from symbolic to neurosymbolic systems: the quest for integration. In: Mascardi and Omicini [136]
2. Agiollo, A., Ciatto, G., Omicini, A.: Graph neural networks as the copula mundi between logic and machine learning: a roadmap. In: Calegari et al. [52], pp. 98–115. http://ceur-ws.org/Vol-2963/paper18.pdf
3. Albertoni, R., Martelli, M., Mascardi, V., Miglia, S.: Specifica, implementazione ed esecuzione di un prototipo di sistema multi-agente in D-CaseLP. In: De Paoli et al. [89], pp. 149–156. http://giuseppevizzari.github.io/WOA-proceedings-archive/pdfs/woa2002/14.pdf
4. Alderighi, M., Baldoni, M., Baroglio, C., Micalizio, R., Tedeschi, S. (eds.): WOA 2024 – 25th Workshop “From Objects to Agents”, CEUR Workshop Proceedings, vol. 3735. Bard, AO, Italy (2024). http://ceur-ws.org/Vol-3735/
5. Alzetta, F., Giorgini, P.: Towards a real-time BDI model for ROS 2. In: Bergenti and Monica [31], pp. 1–7. http://ceur-ws.org/Vol-2404/paper01.pdf
6. Amato, A., Di Martino, B., Venticinque, S.: BDI intelligent agents for augmented exploitation of pervasive environments. In: Fortino et al. [104], pp. 81–88. http://ceur-ws.org/Vol-741/ID16_AmatoDiMartinoVenticinque.pdf
7. Ancona, D., Briola, D., Ferrando, A., Mascardi, V.: MAS-DRiVe: a practical approach to decentralized runtime verification of agent interaction protocols. In: Santoro et al. [170], pp. 35–43. http://ceur-ws.org/Vol-1664/w7.pdf
8. Ancona, D., Ferrando, A., Mascardi, V.: Agents interoperability via conformance modulo mapping. In: Cossentino et al. [80], pp. 109–115. http://ceur-ws.org/Vol-2215/paper_18.pdf
9. Ancona, D., Franceschini, L., Ferrando, A., Mascardi, V.: RML: theory and practice of a domain specific language for runtime verification. Sci. Comput. Program. **205**, 102610 (2021)
10. Armando, A., Zini, D.: Towards interoperable mechanized reasoning systems: the logic broker architecture. In: Corradi et al. [73], pp. 70–75. http://lia.disi.unibo.it/books/woa00/pdf/14.pdf
11. Armano, G., Paoli, F.D., Omicini, A., Vargiu, E. (eds.): WOA 2003 – 4th Workshop “From Objects to Agents”. Pitagora Editrice Bologna, Villasimius, CA, Italy (2003). http://giuseppevizzari.github.io/WOA-proceedings-archive/woa-2003.html
12. Baldoni, M., Baroglio, C.: Some thoughts about commitment protocols (position paper). In: De Paoli and Vizzari [90], pp. 68–71. http://ceur-ws.org/Vol-892/paper11.pdf
13. Baldoni, M., et al.: MERCURIO: an interaction-oriented framework for designing, verifying and programming multi-agent systems. In: Omicini and Viroli [146], pp. 87–94. http://ceur-ws.org/Vol-621/paper13.pdf
14. Baldoni, M., Baroglio, C., Bergenti, F., Garro, A. (eds.): WOA 2013 – 14th Workshop “From Objects to Agents”, CEUR Workshop Proceedings, vol. 1099. Turin, Italy (2013). http://ceur-ws.org/Vol-1099/
15. Baldoni, M., Baroglio, C., Berio, G., Marengo, E.: Declarative representation of curricula models: an LTL-and UML-based approach. In: Baldoni et al. [20], pp. 34–41. http://woa07.dibris.unige.it/papers/BaldoniDeclarative.pdf
16. Baldoni, M., Baroglio, C., Marengo, E., Patti, V., Capuzzimati, F.: Learn the rules so you know how to break them properly. In: Fortino et al. [104], pp. 11–18. http://ceur-ws.org/Vol-741/ID1_BaldoniBaroglioMarengoPattiCapuzzimati.pdf

17. Baldoni, M., Baroglio, C., Martelli, A., Patti, V.: Conformance and interoperability in open enviroments. In: De Paoli et al. [88], pp. 151–157. http://ceur-ws.org/Vol-204/P11.pdf
18. Baldoni, M., Baroglio, C., Martelli, A., Patti, V., Schifanella, C., Torasso, L., Mascardi, V.: Personalization, verification and conformance for logic-based communicating agents. In: Corradini et al. [75], pp. 177–183. http://lia.deis.unibo.it/books/woa2005/papers/25.pdf
19. Baldoni, M., Baroglio, C., Patti, V., Schifanella, C.: Conservative re-use ensuring matches for service selection. In: Baldoni et al. [22], pp. 28–36. http://www.pa.icar.cnr.it/woa08/materiali/paper/paper_8.pdf
20. Baldoni, M., Boccalatte, A., De Paoli, F., Martelli, M., Mascardi, V. (eds.): WOA 2007 – 8th Workshop "From Objects to Agents". Seneca Edizioni Torino, Genova, Italy (2007). http://woa07.disi.unige.it/ProceedingsWOA2007.zip
21. Baldoni, M., Boella, G., van der Torre, L.W.N.: Social roles, from agents back to objects. In: Corradini et al. [75], pp. 164–170. http://lia.deis.unibo.it/books/woa2005/papers/23.pdf
22. Baldoni, M., Cossentino, M., De Paoli, F., Seidita, V. (eds.): WOA 2008 – 9th Workshop "From Objects to Agents". Seneca Edizioni Torino, Palermo, Italy (2008). http://www.pa.icar.cnr.it/woa08/materiali/Proceedings.pdf
23. Baldoni, M., De Paoli, F., Martelli, A., Omicini, A. (eds.): WOA 2004 – 5th Workshop "From Objects to Agents". Pitagora Editrice Bologna, Torino, Italy (2004). http://lia.deis.unibo.it/books/woa2004/atti.pdf
24. Bella, G., Cantone, D., Nicolosi Asmundo, M., Santamaria, D.F.: The ontology for agents, systems and integration of services: Recent advancements of OASIS. In: Ferrando and Mascardi [101], pp. 176–193. http://ceur-ws.org/Vol-3261/paper14.pdf
25. Bellamy, R.K., Dey, K., Hind, M., Hoffman, S.C., Houde, S., Kannan, K., Lohia, P., Martino, J., Mehta, S., Mojsilović, A., et al.: AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. IBM J. Res. Dev. **63**(4/5), 4–1 (2019)
26. Bellifemine, F.L., Caire, G., Greenwood, D.: Developing Multi-Agent Systems with JADE. John Wiley & Sons (2007)
27. Beretta, D., Monica, S., Bergenti, F.: Capturing a recursive pattern in neural-symbolic reinforcement learning. In: Falcone et al. [98], pp. 17–31. http://ceur-ws.org/Vol-3579/paper2.pdf
28. Bergenti, F. (ed.): WOA 2009 – 10th Workshop "From Objects to Agents". Seneca Edizioni Torino, Parma, Italy (2009). http://www.ailab.unipr.it/woa09/papers/
29. Bergenti, F., Caire, G., Monica, S., Poggi, A.: The first twenty years of agent-based software development with JADE. Auton. Agents Multi-Agent Syst. **34**(36) (2020). https://doi.org/10.1007/s10458-020-09460-z
30. Bergenti, F., Iotti, E., Monica, S., Poggi, A.: A case study of the JADEL programming language. In: Santoro et al. [170], pp. 85–90. http://ceur-ws.org/Vol-1664/w15.pdf
31. Bergenti, F., Monica, S. (eds.): WOA 2019 – 20th Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 2404. Parma, Italy (2019). http://ceur-ws.org/Vol-2404/
32. Bergenti, F., Monica, S., Petrosino, G.: A comprehensive presentation of the Jadescript agent-oriented programming language. In: EUMAS. Lecture Notes in Computer Science, vol. 14282, pp. 100–115. Springer, Cham (2023)
33. Bergenti, F., Petrosino, G.: Overview of a scripting language for JADE-based multi-agent systems. In: Cossentino et al. [80], pp. 57–62. http://ceur-ws.org/Vol-2215/paper_10.pdf

34. Boissier, O., Bordini, R.H., Hübner, J.F., Ricci, A.: Multi-Agent Oriented Programming: Programming Multi-Agent Systems Using JaCaMo. The MIT Press (2020). https://mitpress.mit.edu/9780262044578/multi-agent-oriented-programming/
35. Boissier, O., Bordini, R.H., Hübner, J.F., Ricci, A., Santi, A.: Multi-agent oriented programming with JaCaMo. Sci. Comput. Program. **78**(6), 747–761 (2013). https://doi.org/10.1016/j.scico.2011.10.004
36. Bordini, R.H., Costantini, S., Monaldini, A., Vozna, A.: From pure Prolog to logic agent-oriented programming languages. In: Alderighi et al. [4], pp. 271–285. http://ceur-ws.org/Vol-3735/paper_20.pdf
37. Bordini, R.H., El Fallah Seghrouchni, A., Hindriks, K.V., Logan, B., Ricci, A.: Agent programming in the cognitive era. Auton. Agents Multi-Agent Syst. **34**(37) (2020). https://doi.org/10.1007/s10458-020-09453-y
38. Bordini, R.H., Hübner, J.F., Wooldridge, M.J.: Programming Multi-Agent Systems in AgentSpeak using Jason. Wiley (2007). https://www.wiley.com/Programming+Multi-Agent+Systems+in+AgentSpeak+using+Jason-p-9780470061831
39. Bouquet, P., Kuper, G.M., Zanobini, S.: Asking and answering queries semantically. In: Corradini et al. [75], pp. 22–27. http://lia.deis.unibo.it/books/woa2005/papers/4.pdf
40. Briola, D., Locoro, A., Mascardi, V.: Ontology agents in FIPA-compliant platforms: a survey and a new proposal. In: Baldoni et al. [22], pp. 68–75. http://www.pa.icar.cnr.it/woa08/materiali/paper/paper_7.pdf
41. Briola, D., Mascardi, V.: Design and implementation of a NetLogo interface for the stand-alone FYPA system. In: Fortino et al. [104], pp. 41–50. http://ceur-ws.org/Vol-741/ID10_Briola_Mascardi.pdf
42. Briola, D., Mascardi, V.: Multi agent resource allocation: a comparison of five negotiation protocols. In: Fortino et al. [104], pp. 95–104. http://ceur-ws.org/Vol-741/ID9_Briola_Mascardi.pdf
43. Briola, D., Mascardi, V., Gioseffi, M.: OntologyBeanGenerator 5.0: Extending ontology concepts with methods and exceptions. In: Cossentino et al. [80], pp. 116–123. http://ceur-ws.org/Vol-2215/paper_19.pdf
44. Briola, D., Mascardi, V., Martelli, M., Arecco, G., Caccia, R., Milani, C.: A Prolog-based MAS for railway signalling monitoring: Implementation and experiments. In: Baldoni et al. [22], pp. 11–18, http://www.pa.icar.cnr.it/woa08/materiali/paper/paper_4.pdf
45. Briola, D., Mascardi, V., Martelli, M., Caccia, R., Milani, C.: Dynamic resource allocation in a MAS: A case study from the industry. In: Bergenti [28], pp. 125–133. http://www.ailab.unipr.it/woa09/papers/Briola.pdf
46. Busetta, P., Dragoni, M.: Composing cognitive agents from behavioural models in PRESTO. In: Di Napoli et al. [93], pp. 85–90. http://ceur-ws.org/Vol-1382/paper13.pdf
47. Busetta, P., Ghidini, C., Pedrotti, M., De Angeli, A., Menestrina, Z., et al.: Briefing virtual actors: a first report on the PRESTO project. In: Proceedings of the AI and Games Symposium at AISB (2014)
48. Cabitza, F., Locatelli, M.P., Sarini, M.: CASMAS: an agent-based support for modulated participation in cooperative applications. In: Corradini et al. [75], pp. 111–119. http://lia.deis.unibo.it/books/woa2005/papers/16.pdf
49. Caire, G., Gotta, D., Banzi, M.: WADE: a software platform to develop mission critical applications exploiting agents and workflows. In: AAMAS (Industry Track), pp. 29–36. IFAAMAS (2008)

50. Calegari, R., Ciatto, G., Dellaluce, J., Omicini, A.: Interpretable narrative explanation for ML predictors with LP: a case study for XAI. In: Bergenti and Monica [31], pp. 105–112. http://ceur-ws.org/Vol-2404/paper16.pdf
51. Calegari, R., Ciatto, G., Denti, E., Omicini, A., Sartor, G. (eds.): WOA 2020 – 21st Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 2706. Bologna, Italy (2020). http://ceur-ws.org/Vol-2706/
52. Calegari, R., Ciatto, G., Denti, E., Omicini, A., Sartor, G. (eds.): WOA 2021 – 22nd Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 2963. Bologna, Italy (2021). http://ceur-ws.org/Vol-2963/
53. Calegari, R., Ciatto, G., Mariani, S., Denti, E., Omicini, A.: Logic programming in space-time: The case of situatedness in LPaaS. In: Cossentino et al. [80], pp. 63–68. http://ceur-ws.org/Vol-2215/paper_11.pdf
54. Calegari, R., Ciatto, G., Mascardi, V., Omicini, A.: Logic-based technologies for multi-agent systems: a systematic literature review. Auton. Agents Multi-Agent Syst. **35**(1) (2021). https://doi.org/10.1007/s10458-020-09478-3
55. Calegari, R., Denti, E., Mariani, S., Omicini, A.: Towards logic programming as a service: Experiments in tuProlog. In: Santoro et al. [170], pp. 79–84. http://ceur-ws.org/Vol-1664/w14.pdf
56. Calegari, R., Omicini, A., Sartor, G.: Computable law as argumentation-based MAS. In: Calegari et al. [51], pp. 54–68. http://ceur-ws.org/Vol-2706/paper10.pdf
57. Cannata, N., Corradini, F., Piersigilli, F., Merelli, E., Vito, L.: Semantic resource management in MAS. In: Baldoni et al. [20], pp. 42–47. http://woa07.disi.unige.it/papers/CannataSemantic.pdf
58. Cantone, D., Longo, C.F., Nicolosi Asmundo, M., Santamaria, D.F., Santoro, C.: Towards an ontology-based framework for a behavior-oriented integration of the IoT. In: Bergenti and Monica [31], pp. 119–126. http://ceur-ws.org/Vol-2404/paper18.pdf
59. Cantucci, F., Falcone, R.: A computational model for cognitive human-robot interaction: An approach based on theory of delegation. In: Bergenti and Monica [31], pp. 127–133. http://ceur-ws.org/Vol-2404/paper19.pdf
60. Cantucci, F., Falcone, R.: Autonomous critical help provided by an artificial agent in the field of cultural heritage. In: Ferrando and Mascardi [101], pp. 152–163. http://ceur-ws.org/Vol-3261/paper12.pdf
61. Cantucci, F., Falcone, R.: A cognitive approach to model intelligent collaboration in human-robot interaction. In: Falcone et al. [98], pp. 138–150. http://ceur-ws.org/Vol-3579/paper10.pdf
62. Cardoso, R.C., Ferrando, A.: A review of agent-based programming for multi-agent systems. Comput. **10**(2), 16 (2021)
63. Casadei, M., Viroli, M.: A framework to specify and verify computational fields for pervasive computing systems. In: De Paoli and Vizzari [90], pp. 72–81. http://ceur-ws.org/Vol-892/paper2.pdf
64. Castelfranchi, C., Chella, A., Falcone, R., Lanza, F., Seidita, V.: Endowing robots with self-modeling abilities for trustful human-robot interactions. In: Bergenti and Monica [31], pp. 22–28. http://ceur-ws.org/Vol-2404/paper04.pdf
65. Catta, D., Leneutre, J., Malvone, V.: Subset sabotage games & attack graphs. In: Ferrando and Mascardi [101], pp. 209–218. http://ceur-ws.org/Vol-3261/paper16.pdf
66. Chella, A., Lanza, F., Seidita, V.: Representing and developing knowledge using Jason, Cartago and OWL. In: Cossentino et al. [80], pp. 147–152. http://ceur-ws.org/Vol-2215/paper_23.pdf

67. Chesani, F., et al.: Protocol specification and verification by using computational logic. In: Corradini et al. [75], pp. 184–192. http://lia.deis.unibo.it/books/woa2005/papers/26.pdf
68. Ciampolini, A., Lamma, E., Mello, P., Torroni, P.: Expressing collaboration and competition among abductive logic agents. In: Corradi et al. [73], pp. 64–69. http://giuseppevizzari.github.io/WOA-proceedings-archive/pdfs/woa2000/WOA13.pdf
69. Ciatto, G., Calegari, R., Mariani, S., Denti, E., Omicini, A.: From the blockchain to logic programming and back: research perspectives. In: Cossentino et al. [80], pp. 69–74. http://ceur-ws.org/Vol-2215/paper_12.pdf
70. Ciatto, G., Calegari, R., Siboni, E., Denti, E., Omicini, A.: 2P-Kt: logic programming with objects & functions in Kotlin. In: Calegari et al. [51], pp. 219–236. http://ceur-ws.org/Vol-2706/paper14.pdf
71. Cordì, V., Lombardi, P., Martelli, M., Mascardi, V.: An ontology-based similarity between sets of concepts. In: Corradini et al. [75], pp. 16–21. http://lia.deis.unibo.it/books/woa2005/papers/3.pdf
72. Corradi, A., Montanari, R., Stefanelli, C.: Mobile agents integrity in e-commerce applications. In: Proceedings of 19th IEEE International Conference on Distributed Computing Systems. Workshops on Electronic Commerce and Web-based Applications. Middleware, pp. 59–64 (1999). https://doi.org/10.1109/ECMDD.1999.776415
73. Corradi, A., Omicini, A., Poggi, A. (eds.): WOA 2000 – 1st Workshop "From Objects to Agents", Atti di Congressi, vol. 1195. Pitagora Editrice Bologna, Parma, Italy (2000). http://giuseppevizzari.github.io/WOA-proceedings-archive/woa-2000.html
74. Corradini, F., Culmone, R., Di Berardini, M.R., Merelli, E.: Integrating ontologies in mobile agents. In: Corradini et al. [75], pp. 37–45. http://lia.deis.unibo.it/books/woa2005/papers/6.pdf
75. Corradini, F., De Paoli, F., Merelli, E., Omicini, A. (eds.): WOA 2005 – 6th Workshop "From Objects to Agents". Pitagora Editrice Bologna, Camerino, MC, Italy (2005). http://lia.deis.unibo.it/books/woa2005/atti.pdf
76. Cossentino, M., Lodato, C., Lopes, S., Ribino, P., Seidita, V., Chella, A.: A UML-based notation for representing MAS organizations. In: Fortino et al. [104], pp. 133–139. http://ceur-ws.org/Vol-741/ID20_CossentinoLodatoLopesRibiniSeiditaChella.pdf
77. Cossentino, M., Lopes, S., Nuzzo, A., Renda, G., Sabatucci, L.: A comparison of the basic principles and behavioural aspects of Akka, JaCaMo and Jade development frameworks. In: Cossentino et al. [80], pp. 133–141. http://ceur-ws.org/Vol-2215/paper_21.pdf
78. Cossentino, M., Lopes, S., Renda, G., Sabatucci, L., Zaffora, F.: A metamodel of a multi-paradigm approach to smart cyber-physical systems development. In: Bergenti, Monica (eds.) [31], pp. 35–41. http://ceur-ws.org/Vol-2404/paper06.pdf
79. Cossentino, M., et al.: GIMT: A tool for ontology and goal modeling in BDI multi-agent design. In: Santoro and Bergenti [169], pp. 81–88. http://ceur-ws.org/Vol-1260/paper10.pdf
80. Cossentino, M., Sabatucci, L., Seidita, V. (eds.): WOA 2018 – 19th Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 2215. Palermo, Italy (2018). http://ceur-ws.org/Vol-2215/

81. Costantini, S., Dell'Acqua, P., Tocchio, A.: Expressing preferences declaratively in logic-based agent languages. In: De Paoli et al. [88], pp. 138–143. http://ceur-ws.org/Vol-204/P05.pdf
82. Costantini, S., Formisano, A., Pitoni, V.: A timed epistemic logic for formalizing cooperation among groups of agents. In: Falcone et al. [98], pp. 151–166. http://ceur-ws.org/Vol-3579/paper11.pdf
83. Costantini, S., Tocchio, A.: Strips-like planning in the DALI logic programmming language. In: Armano et al. [11], pp. 115–120. http://giuseppevizzari.github.io/WOA-proceedings-archive/pdfs/woa2003/05.pdf
84. Costantini, S., Tocchio, A.: Learning by knowledge exchange in logical agents. In: Corradini et al. [75], pp. 1–8. http://lia.deis.unibo.it/books/woa2005/papers/1.pdf
85. Cranefield, S., Finin, T.W., Tamma, V.A.M., Willmott, S. (eds.): Ontologies in Agent Systems 2003, Proceedings of the Workshop on Ontologies in Agent Systems (OAS 2003) at the 2nd International Joint Conference on Autonomous Agents and Multi-Agent Systems, Melbourne, Australia, July 15, 2003, CEUR Workshop Proceedings, vol. 73. CEUR-WS.org (2003). https://ceur-ws.org/Vol-73/
86. Cristani, M., Olivieri, F., Governatori, G., Buriola, G.: Simulating the law in a multi-agent system. In: Alderighi et al. [4], pp. 236–251. http://ceur-ws.org/Vol-3735/paper_18.pdf
87. D'Asaro, F.A., Origlia, A., Rossi, S.: Towards a logic-based approach for multimodal fusion and decision making during motor rehabilitation sessions. In: Bergenti and Monica [31], pp. 8–13. http://ceur-ws.org/Vol-2404/paper02.pdf
88. De Paoli, F., Di Stefano, A., Omicini, A., Santoro, C. (eds.): WOA 2006 – 7th Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 204. Catania, Italy (2006). http://ceur-ws.org/Vol-204/
89. De Paoli, F., Manzoni, S., Poggi, A. (eds.): WOA 2002 – 3rd Workshop "From Objects to Agents". Pitagora Editrice Bologna, Milano, Italy (2002). http://giuseppevizzari.github.io/WOA-proceedings-archive/woa-2002.html
90. De Paoli, F., Vizzari, G. (eds.): WOA 2012 – 13th Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 892. Milano, Italy (2012). http://ceur-ws.org/Vol-892/
91. Delzanno, G., Martelli, M.: Proofs as computations in linear logic. In: APPIA-GULP-PRODE, pp. 155–166 (1996)
92. Deufemia, V., Polese, G., Tortora, G., Vacca, M.: Conceptual foundations of interrogative agents. In: Baldoni et al. [20], pp. 26–33. http://woa07.disi.unige.it/papers/DeufemiaConceptual.pdf
93. Di Napoli, C., Rossi, S., Staffa, M. (eds.): WOA 2015 – 16th Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 1382. Naples, Italy (2015). http://ceur-ws.org/Vol-1382/
94. Dignum, V.: Responsible artificial intelligence: how to develop and use AI in a responsible way. Springer Nature (2019)
95. Esfahani, Z.N., Engelmann, D.C., Ferrando, A., Margarone, M., Mascardi, V.: Integrating ontologies and cognitive conversational agents in On2Conv. In: Multi-Agent Systems: 20th European Conference, EUMAS 2023, Naples, Italy, September 14–15, 2023, Proceedings, Lecture Notes in Computer Science, vol. 14282, pp. 66–82. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-43264-4_5
96. European Commission: Ethics guidelines for trustworthy AI (2019). https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai

97. Falcone, R., Castelfranchi, C.: Transitivity in trust: a discussed property. In: Omicini and Viroli [146], pp. 155–160. http://ceur-ws.org/Vol-621/paper22.pdf
98. Falcone, R., Castelfranchi, C., Sapienza, A., Cantucci, F. (eds.): WOA 2023 – 24th Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 3579. Roma, Italy (2023). http://ceur-ws.org/Vol-3579/
99. Ferrando, A., Gatti, A., Mascardi, V.: Geometric and spatial reasoning in BDI agents: a survey. In: CILC. CEUR Workshop Proceedings, vol. 3733. CEUR-WS.org (2024)
100. Ferrando, A., Malvone, V.: Hands-on VITAMIN: A compositional tool for model checking of multi-agent systems. In: Alderighi et al. [4], pp. 157–169. http://ceur-ws.org/Vol-3735/paper_12.pdf
101. Ferrando, A., Mascardi, V. (eds.): WOA 2022 – 23rd Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 3261. Genova, Italy (2022). http://ceur-ws.org/Vol-3261/
102. Fichera, L., Marletta, D., Nicosia, V., Santoro, C.: A methodology to extend imperative languages with AgentSpeak declarative constructs. In: Omicini and Viroli [146], pp. 30–38. http://ceur-ws.org/Vol-621/paper05.pdf
103. Fisher, M., Mascardi, V., Rozier, K.Y., Schlingloff, B., Winikoff, M., Yorke-Smith, N.: Towards a framework for certification of reliable autonomous systems. Auton Agents Multi-Agent Syst. textbf35(8) (2021). https://doi.org/10.1007/s10458-020-09487-2
104. Fortino, G., Garro, A., Palopoli, L., Russo, W., Spezzano, G. (eds.): WOA 2011 – 12th Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 741. Rende, Italy (2011). http://ceur-ws.org/Vol-741/
105. Gatti, A., Mascardi, V.: VEsNA, a framework for virtual environments via natural language agents and its application to factory automation. Robotics **12**(2), 46 (2023)
106. Giannotti, F., Naretto, F., Bodria, F.: Explainable for trustworthy AI. In: Chetouani, M., Dignum, V., Lukowicz, P., Sierra, C. (eds.) Human-Centered Artificial Intelligence: Advanced Lectures, pp. 175–195. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-24349-3_10
107. Giordano, L., Martelli, A., Terenziani, P., Bottrighi, A., Montani, S.: A temporal approach to the specification and verification of interaction protocols. In: Corradini et al. [75], pp. 171–176. http://lia.deis.unibo.it/books/woa2005/papers/24.pdf
108. Grimaldi, C., Rossi, S.: Towards transparent computational models of theory of mind in collaborative environments. In: Alderighi et al. [4], pp. 73–83. http://ceur-ws.org/Vol-3735/paper_06.pdf
109. Hendler, J.A.: Agents and the semantic web. IEEE Intell. Syst. **16**(2), 30–37 (2001)
110. Hinton, A., Kwiatkowska, M.Z., Norman, G., Parker, D.: PRISM: a tool for automatic verification of probabilistic systems. In: Hermanns, H., Palsberg, J. (eds.) TACAS. Lecture Notes in Computer Science, vol. 3920, pp. 441–444. Springer, Cham (2006). https://doi.org/10.1007/11691372_29
111. Hübner, J.F., Sichman, J.S., Boissier, O.: Developing organised multiagent systems using the MOISE^{+} model: programming issues at the system and agent levels. Int. J. Agent Oriented Softw. Eng. **1**(3/4), 370–395 (2007). https://doi.org/10.1504/IJAOSE.2007.016266
112. Ivan, D., Ferrando, A., Gatti, A., Guerrini, G., Mascardi, V.: Integrating procedural ontologies in VEsNA: Study, requirements, and preliminary design. In: Alderighi et al. [4], pp. 186–199. http://ceur-ws.org/Vol-3735/paper_14.pdf

113. Jennings, N.R., Sycara, K.P., Wooldridge, M.J.: A roadmap of agent research and development. Auton. Agents Multi-Agent Syst. **1**, 7–38 (1998). https://doi.org/10.1023/A:1010090405266
114. Juan, T., Sterling, L., Martelli, M., Mascardi, V.: Customizing AOSE methodologies by reusing AOSE features. In: AAMAS, pp. 113–120. ACM (2003)
115. Kouvaros, P., Lomuscio, A., Pirovano, E., Punchihewa, H.: Formal verification of open multi-agent systems. In: AAMAS '19: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent System, pp. 179–187. International Foundation for Autonomous Agents and Multiagent Systems (2019). https://dl.acm.org/doi/10.5555/3306127.3331691
116. Kowalski, R.A.: Logical foundations for multi-agent systems. In: Alpuente, M., Sessa, M.I. (eds.) 1995 Joint Conference on Declarative Programming, GULP-PRODE'95, Marina di Vietri, Italy, September 11–14, 1995, pp. 39–40 (1995)
117. Kowalski, R.A.: Using meta-logic to reconcile reactive with rational agents. In: Crabtree, I.B., Jennings, N.R. (eds.) Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology, PAAM 1996, Westminster Central Hall, London, UK, April 22–24, 1996, pp. 361–374. Practical Application Company Ltd. (1996)
118. Kowalski, R.A., Sadri, F.: Towards a unified agent architecture that combines rationality with reactivity. In: Pedreschi, D., Zaniolo, C. (eds.) Logic in Databases. Lecture Notes in Computer Science, vol. 1154, pp. 137–149. Springer, Cham (1996). https://doi.org/10.1007/BFb0031739
119. Locoro, A., Mascardi, V., Mortara, F., Sanna, R.: Managing unavailabilities in a dynamic scenario following an agent-based approach. In: Fortino et al. [104], pp. 58–67. http://ceur-ws.org/Vol-741/ID17_LocoroMascardiMontaraSanna.pdf
120. Longo, C.F., Longo, F., Santoro, C.: A reactive cognitive architecture based on natural language processing for the task of decision-making using a rich semantic. In: Calegari et al. [51], pp. 201–218. http://ceur-ws.org/Vol-2706/paper2.pdf
121. Longo, C.F., Santoro, C., Nicolosi Asmundo, M., Santamaria, D.F., Cantone, D.: SW-CASPAR: Reactive-cognitive architecture based on natural language processing for the task of decision-making in the open-world assumption. In: Calegari et al. [52], pp. 178–193. http://ceur-ws.org/Vol-2963/paper10.pdf
122. Longo, F., Santoro, C.: A Python-based assistant agent able to interact with natural language. In: Cossentino et al. [80], pp. 142–146. http://ceur-ws.org/Vol-2215/paper_22.pdf
123. Loseto, G., Ruta, M., Scioscia, F., Sciascio, E.D., Mongiello, M.: Mining the user profile from a smartphone: a multimodal agent framework. In: Baldoni et al. [14], pp. 47–53. http://ceur-ws.org/Vol-1099/paper10.pdf
124. Maccagnan, A., Vardanega, T., Feltrin, E., Valle, G., Riva, M., Cannata, N.: A multi-agent system for the automated handling of experimental protocols in biological laboratories. In: Omicini and Viroli [146], pp. 126–132. http://ceur-ws.org/Vol-621/paper18.pdf
125. Maggi, P., Sisto, R.: Experiments on formal verification of mobile agent data integrity properties. In: De Paoli et al. [89], pp. 131–136. http://giuseppevizzari.github.io/WOA-proceedings-archive/pdfs/woa2002/15.pdf
126. Magnolo, E., Manenti, L., Manzoni, S., Sartori, F.: Towards a MAS model for crowd simulation at pop-rock concerts exploiting ontologies and fuzzy logic. In: Bergenti [28], pp. 92–99. http://www.ailab.unipr.it/woa09/papers/Magnolo.pdf
127. Marguglio, A., Cammarata, G., Bonura, S., Francaviglia, G., Puccio, M., Morreale, V.: Design and development of intentional systems with PRACTIONIST Studio. In: Baldoni et al. [22], pp. 37–45. http://www.pa.icar.cnr.it/woa08/materiali/paper/paper_16.pdf

128. Mariani, S., Omicini, A.: Multi-paradigm coordination for MAS: integrating heterogeneous coordination approaches in MAS technologies. In: Santoro et al. [170], pp. 91–99. http://ceur-ws.org/Vol-1664/w16.pdf
129. Marini, S., Martelli, M., Mascardi, V., Zini, F.: HEMASL: a flexible language to specify heterogeneous agents. In: Corradi et al. [73], pp. 76–81. http://giuseppevizzari.github.io/WOA-proceedings-archive/pdfs/woa2000/WOA15.pdf
130. Martelli, M., Mascardi, V., Zini, F.: Towards multi-agent software prototyping. In: PAAM, pp. 331–354. Practical Application Company Ltd. (1998)
131. Martino, S.D., Asmundo, M.N., Rizzo, S.A., Santamaria, D.F.: Modeling the video game environment: the VideOWL ontology. In: Falcone et al. [98], pp. 191–205. http://ceur-ws.org/Vol-3579/paper15.pdf
132. Mascardi, V., et al.: The Indiana MAS project: goals and preliminary results. In: De Paoli and Vizzari [90], pp. 55–62. http://ceur-ws.org/Vol-892/paper10.pdf
133. Mascardi, V., Cordì, V., Rosso, P.: A comparison of upper ontologies. In: Baldoni et al. [20], pp. 55–64. http://woa07.dibris.unige.it/papers/mascardi.pdf
134. Mascardi, V., Demergasso, D., Ancona, D.: Languages for programming BDI-style agents: an overview. In: Corradini et al. [75], pp. 9–15. http://lia.deis.unibo.it/books/woa2005/papers/2.pdf
135. Mascardi, V., Hendler, J.A., Papaleo, L.: Semantic web and declarative agent languages and technologies: current and future trends - (position paper). In: Baldoni, M., Dennis, L., Mascardi, V., Vasconcelos, W. (eds.) DALT. Lecture Notes in Computer Science, vol. 7784, pp. 197–202. Springer, Cham (2012). https://doi.org/10.1007/978-3-642-37890-4_12
136. Mascardi, V., Omicini, A. (eds.): The Agents Journey: Twenty-five Years of Multi-agent Systems at WOA. Lecture Notes in Computer Science – State-of-the-Art Surveys, Springer, Cham (2026)
137. Messina, F., Santoro, C., Santoro, F.F.: A declarative C++ agent platform for agent-based edge computing. In: Falcone et al. [98], pp. 206–215. http://ceur-ws.org/Vol-3579/paper16.pdf
138. Minotti, M., Santi, A., Ricci, A.: Developing web client applications with JaCa-Web. In: Omicini and Viroli [146], pp. 72–79. http://ceur-ws.org/Vol-621/paper11.pdf
139. Montaldo, E., Sacile, R., Boccalatte, A.: Rule-based agents for workflow applications in manifacturing information systems. In: Omicini and Viroli [145], pp. 90–94. http://giuseppevizzari.github.io/WOA-proceedings-archive/pdfs/woa2001/pdf/16.pdf
140. Morreale, V., Bonura, S., Centineo, F., Rossi, A., Cossentino, M., Gaglio, S.: PRACTIONIST: implementing PRACTIcal reasONIng sySTems. In: Corradini et al. [75], pp. 66–74. http://lia.deis.unibo.it/books/woa2005/papers/10.pdf
141. Morreale, V., Bonura, S., Francaviglia, G., Centineo, F., Cossentino, M., Gaglio, S.: Reasoning about goals in BDI agents: The PRACTIONIST framework. In: De Paoli et al. [88], pp. 187–194. http://ceur-ws.org/Vol-204/P20.pdf
142. Morreale, V., et al.: PRACTIONIST: a framework for developing BDI agent systems. In: De Paoli et al. [88], pp. 4–5. http://ceur-ws.org/Vol-204/D02.pdf
143. Oliva, E., Viroli, M., Omicini, A.: Minority game: a logic-based approach in TuCSoN. In: De Paoli et al. [88], pp. 181–186. http://ceur-ws.org/Vol-204/P02.pdf
144. Oliva, E., Viroli, M., Omicini, A.: Arguments and artifacts for dispute resolution. In: Baldoni et al. [22], pp. 46–53. http://www.pa.icar.cnr.it/woa08/materiali/paper/paper_12.pdf
145. Omicini, A., Viroli, M. (eds.): WOA 2001 – 2nd Workshop "From Objects to Agents". Pitagora Editrice Bologna, Modena, Italy (2001). http://giuseppevizzari.github.io/WOA-proceedings-archive/woa-2001.html

146. Omicini, A., Viroli, M. (eds.): WOA 2010 – 11th Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 621. Rimini, Italy (2010). http://ceur-ws.org/Vol-621/
147. Omicini, A., Zambonelli, F.: Co-ordination of mobile information agents in TuCSoN. Internet Res. **8**(5), 400–413 (1998). https://doi.org/10.1108/10662249810241266
148. Passadore, A., Grosso, A., Boccalatte, A.: Indexing enterprise knowledge bases with AgentSeeker. In: Bergenti [28], pp. 69–75. http://www.ailab.unipr.it/woa09/papers/Passadore.pdf
149. Petrosino, G., Bergenti, F.: Extending message handlers with pattern matching in the Jadescript programming language. In: Bergenti and Monica [31], pp. 113–118. http://ceur-ws.org/Vol-2404/paper17.pdf
150. Pisano, G., Calegari, R., Omicini, A.: Towards cooperative argumentation for MAS: an actor-based approach. In: Calegari et al. [52], pp. 162–177. http://ceur-ws.org/Vol-2963/paper17.pdf
151. Pisano, G., Calegari, R., Omicini, A., Sartor, G.: Arg-tuProlog: a tuProlog-based argumentation framework. In: CILC. CEUR Workshop Proceedings, vol. 2710, pp. 51–66. CEUR-WS.org (2020)
152. Pisano, G., Ciatto, G., Calegari, R., Omicini, A.: Neuro-symbolic computation for XAI: towards a unified model. In: Calegari et al. [51], pp. 101–117. http://ceur-ws.org/Vol-2706/paper18.pdf
153. Piunti, M., Ricci, A., Boissier, O., Hübner, J.F.: Programming open systems with agents, environments and organizations. In: Omicini and Viroli [146], pp. 39–47. http://ceur-ws.org/Vol-621/paper06.pdf
154. Piunti, M., Ricci, A., Santi, A.: SOA/WS applications using cognitive agents working in CArtAgO environments. In: Bergenti [28], pp. 116–124. http://www.ailab.unipr.it/woa09/papers/Piunti.pdf
155. Poggi, A., Tomaiuolo, M., Turci, P.: An agent-based service oriented architecture. In: Baldoni et al. [20], pp. 157–165. http://woa07.disi.unige.it/papers/PoggiSOA.pdf
156. Rafanelli, A., Costantini, S., De Gasperis, G.: A multi-agent-system framework for flooding events. In: Ferrando and Mascardi [101], pp. 142–151. http://ceur-ws.org/Vol-3261/paper11.pdf
157. Rao, A.S.: AgentSpeak(L): BDI agents speak out in a logical computable language. In: Van de Velde, W., Perram, J.W. (eds.) European workshop on modelling autonomous agents in a multi-agent world, pp. 42–55. Springer, Cham (1996). https://doi.org/10.1007/BFb0031845
158. Rao, A.S., Georgeff, M.P.: BDI agents: From theory to practice. In: ICMAS, pp. 312–319. The MIT Press (1995)
159. Rawal, A., McCoy, J., Rawat, D.B., Sadler, B.M., Amant, R.S.: Recent advances in trustworthy explainable artificial intelligence: status, challenges, and perspectives. IEEE Trans. Artif. Intell. **3**(6), 852–866 (2022). https://doi.org/10.1109/TAI.2021.3133846
160. Repetto, M., Vecchiola, C., Boccalatte, A.: A knowledge modeling tool for rule-based agents. In: Omicini and Viroli [145], pp. 24–29. http://giuseppevizzari.github.io/WOA-proceedings-archive/pdfs/woa2001/pdf/05.pdf
161. Ribino, P., Cossentino, M., Lodato, C., Lopes, S., Sabatucci, L., Seidita, V.: Ontology and goal model in designing BDI multi-agent systems. In: Baldoni et al. [14], pp. 66–72. http://ceur-ws.org/Vol-1099/paper12.pdf

162. Ricci, A., Piunti, M., Viroli, M., Omicini, A.: Environment programming in CArtAgO. In: El Fallah Seghrouchni, A., Dix, J., Dastani, M., Bordini, R. (eds.) Multi-Agent Programming, Languages, Tools and Applications, pp. 259–288. Springer, Cham (2009). https://doi.org/10.1007/978-0-387-89299-3_8
163. Ricci, A., Viroli, M., Omicini, A.: CArtAgO: A framework for prototyping artifact-based environments in MAS. In: Weyns, D., Parunak, H.V.D., Michel, F. (eds.) Environments for MultiAgent Systems III, Lecture Notes in Computer Science, vol. 4389, chap. 4, pp. 67–86. Springer Berlin Heidelberg (May 2007). https://doi.org/10.1007/978-3-540-71103-2_4, 3rd International Workshop (E4MAS 2006), Hakodate, Japan, 8 May 2006. Selected Revised and Invited Papers
164. Ruta, M., Scioscia, F., Loseto, G., Gramegna, F., Pinto, A., Sciascio, E.D.: Semantic-based social intelligence through multi-agent systems. In: Cossentino et al. [80], pp. 96–102. http://ceur-ws.org/Vol-2215/paper_16.pdf
165. Sabbatini, F., Calegari, R.: Unlocking insights and trust: the value of explainable clustering algorithms for cognitive agents. In: Falcone et al. [98], pp. 232–245. http://ceur-ws.org/Vol-3579/paper18.pdf
166. Sabbatini, F., Ciatto, G., Calegari, R., Omicini, A.: On the design of PSyKE: a platform for symbolic knowledge extraction. In: Calegari et al. [52], pp. 29–48. http://ceur-ws.org/Vol-2963/paper14.pdf
167. Sacile, R., Montaldo, E., Paolucci, M., Boccalatte, A.: Intelligent agents applied to manufacturing: the MAKE-IT approach. In: Corradi et al. [73], pp. 92–97. http://giuseppevizzari.github.io/WOA-proceedings-archive/pdfs/woa2000/WOA18.pdf
168. Santi, A., Guidi, M., Ricci, A.: Exploiting agent-oriented programming for developing Android applications. In: Omicini and Viroli [146], pp. 48–54. http://ceur-ws.org/Vol-621/paper07.pdf
169. Santoro, C., Bergenti, F. (eds.): WOA 2014 – 15th Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 1260. Catania, Italy (2014). http://ceur-ws.org/Vol-1260/
170. Santoro, C., Messina, F., De Benedetti, M. (eds.): WOA 2016 – 17th Workshop "From Objects to Agents", CEUR Workshop Proceedings, vol. 1664. Catania, Italy (2016). http://ceur-ws.org/Vol-1664/
171. Sapienza, A., Falcone, R., Castelfranchi, C.: The positive power of prejudice: A computational model for MAS. In: Di Napoli et al. [93], pp. 39–45. http://ceur-ws.org/Vol-1382/paper6.pdf
172. Schifanella, C., Lusso, L., Baldoni, M., Baroglio, C.: Design and development of a visual environment for writing DyLOG programs. In: Baldoni et al. [23], pp. 43–50. http://giuseppevizzari.github.io/WOA-proceedings-archive/pdfs/woa2004/7.pdf
173. Seidita, V., Lanza, F., Sabella, A.M.P., Chella, A.: Can agents talk about what they are doing? A proposal with Jason and speech acts. In: Ferrando and Mascardi [101], pp. 17–29. http://ceur-ws.org/Vol-3261/paper2.pdf
174. Seidita, V., Sabella, A.M.P., Chella, A.: Agents showing self-disclosure. A preliminary methodological approach. In: Falcone et al. [98], pp. 78–91. http://ceur-ws.org/Vol-3579/paper6.pdf
175. Sguera, S., Stellato, A., Griesi, D., Pazienza, M.T.: J-ALINAs: a JADE-based architecture for linguistic agents. In: De Paoli et al. [88], pp. 83–89. http://ceur-ws.org/Vol-204/P09.pdf
176. Sterling, L.S., Marshall, J.: Humans are not rational; artificial agents are not emotional. In: Ferrando and Mascardi [101], pp. 127–141. http://ceur-ws.org/Vol-3261/paper10.pdf

177. Thiebes, S., Lins, S., Sunyaev, A.: Trustworthy artificial intelligence. Electron. Mark. **31**, 447–464 (2021)
178. Tomaiuolo, M., Bergenti, F., Poggi, A., Turci, P.: OWLBeans–from ontologies to Java classes. In: Baldoni et al. [23], pp. 116–125. http://giuseppevizzari.github.io/WOA-proceedings-archive/pdfs/woa2004/17.pdf
179. Tomaiuolo, M., Turci, P.: Peer-to-peer delegation for accessing web services. In: Omicini and Viroli [146], pp. 65–71. http://ceur-ws.org/Vol-621/paper10.pdf
180. Unniyankal, H., Belardinelli, F., Ferrando, A., Malvone, V.: RMLGym: a formal reward machine framework for reinforcement learning. In: Falcone et al. [98], pp. 1–16. http://ceur-ws.org/Vol-3579/paper1.pdf
181. Vecchiola, C., Grosso, A., Boccalatte, A.: Integrating ontology support within AgentService. In: De Paoli et al. [88], pp. 166–172. http://ceur-ws.org/Vol-204/P22.pdf
182. Vianello, A., Laine, S., Tuomi, E.: Improving trustworthiness of AI solutions: a qualitative approach to support ethically-grounded AI design. Int. J. Human-Comput. Inter. **39**(7), 1405–1422 (2023)
183. Vozna, A., Monaldini, A., Costantini, S., De Gasperis, G., Dell'Acqua, P., Formisano, A., Rafanelli, A.: Evolution of programming languages in agent systems and the role of computational logic. In: Mascardi and Omicini [136]
184. Wickramasinghe, C.S., Marino, D.L., Grandio, J., Manic, M.: Trustworthy AI development guidelines for human system interaction. In: 2020 13th International Conference on Human System Interaction (HSI). pp. 130–136 (2020). https://doi.org/10.1109/HSI49210.2020.9142644
185. Wooldridge, M.J., Jennings, N.R.: Intelligent agents: theory and practice. Knowl. Eng. Rev. **10**(2), 115–152 (1995). https://doi.org/10.1017/S0269888900008122
186. Yan, E., Burattini, S., Hübner, J.F., Ricci, A.: Towards a multi-level explainability framework for engineering and understanding BDI agent systems. In: Falcone et al. [98], pp. 216–231. http://ceur-ws.org/Vol-3579/paper17.pdf
187. Yeung, K.: Recommendation of the council on artificial intelligence (OECD). Int. Leg. Mater. **59**(1), 27–34 (2020). https://doi.org/10.1017/ilm.2020.5