

Engineering Multi-Agent Systems and Generative AI: Report from the Agent Toolkits 2025 Community Session

Andrei Ciortea^{1,2*}, Katharine Beaumont^{3**}, Gianluca Aguzzi⁴, Matteo Baldoni⁵, Cristina Baroglio⁵, Amit K. Chopra⁶, Giovanni Ciatto⁴, Rem Collier³, Mehdi Dastani⁷, Angelo Ferrando⁸, Andrea Gatti⁹, Önder Gürcan¹⁰, Timotheus Kampik^{11,12}, Jérémy Lemée¹, Somsakun Maneerat¹³, Elisa Marengo⁵, Viviana Mascardi⁹, Simon Mayer¹, Roberto Micalizio⁵, Guillaume Muller¹⁴, Vivek Nallur³, Richard Niamke¹³, Andrei Olaru¹⁵, Héloïse Pajot¹⁶, Chloé Petridis¹³, I.S.W.B. Prasetya⁷, Alessandro Ricci⁴, Alexandru Sorici¹⁵, Stefano Tedeschi¹⁷, and Michael Winikoff¹⁸

¹ University of St.Gallen, Switzerland

² Inria, Université Côte d'Azur, I3S, CNRS, Sophia Antipolis, France

³ University College Dublin, Ireland

⁴ University of Bologna, Italy

⁵ Università degli Studi di Torino, Italy

⁶ Lancaster University, UK

⁷ Utrecht University, the Netherlands

⁸ University of Modena-Reggio Emilia, Italy

⁹ University of Genova, Italy

¹⁰ Coral Protocol, London, UK

¹¹ Umeå University, Sweden

¹² SAP, Sweden

¹³ EDF R&D, SEQUOIA, Lab Paris-Saclay, Palaiseau, France

¹⁴ Institut Henri Fayol, ISI dept. Mines Saint-Étienne, France

¹⁵ National University of Science and Technology POLITEHNICA Bucharest, Romania

¹⁶ EDF Innovation Lab, Palo Alto, USA

¹⁷ Università della Valle d'Aosta – Université de la Vallée d'Aoste, Italy

¹⁸ Victoria University of Wellington, New Zealand

Abstract. Intelligent agents have been a cornerstone of Artificial Intelligence (AI) since its early days—and received significant attention in the 1990s when the notion of *autonomous agent* was established, giving rise to research on Engineering Multi-Agent Systems (EMAS). Traditionally, this area has focused on theories, architectures, methodologies, paradigms, and languages for designing, implementing, and governing systems of autonomous agents. More recently, advances in Generative AI—and specifically large language models—have led to a new generation of agents and multi-agent systems, often referred to as Agentic AI.

* Session organizer and corresponding author: andrei.ciortea@unisg.ch

** Session organizer

However, the conceptual bridges, overlaps, and complementarities between Agentic AI and traditional EMAS research are often unclear. The Agent Toolkits 2025 community session invited position statements from senior members of the EMAS community to discuss these recent developments. This paper summarizes and integrates their contributions and outlines key directions for research on EMAS with Generative AI.

Keywords: Engineering Multi-Agent Systems · Generative AI · Agent Toolkits.

1 Introduction

Recent advances in large language models (LLMs) endowed with instruction-following and tool-use capabilities have renewed interest in autonomous agents and multi-agent systems (MAS). As with previous generations of agents, LLM-based agents¹⁹ are often designed for specific tasks, yet their broader potential is realized when they can discover, interact with, and complement one another. Industry-led initiatives and open standards—such as Anthropic’s Model Context Protocol (MCP) [4] or Google’s Agent-to-Agent (A2A) protocol [117]—are rapidly emerging, enabling LLM agents to discover and use tools or to discover and interact within MAS. There is a growing body of research on cognitive architectures for LLM agents (e.g., [116]), on tool use and learning [103, 92, 125, 93, 91], and on communication and interaction (e.g., see [70]). At the same time, the developer community contributes popular articles on design patterns for LLM agents [3], methodologies for developing tools that agents can use effectively [7], and architectures for systems of LLM agents [6]. In this context, it is timely to explore how these recent initiatives can integrate, overlap, or complement the substantial body of research on engineering autonomous agents and MAS—the main objective that motivated the Agent Toolkits 2025 Community Session.

The Agent Toolkits initiative aims to bring together researchers and practitioners working in the domain of Engineering Multi-Agent Systems (EMAS)²⁰ in an open and discussion-oriented setting—in which we welcome the sharing of research results, demonstrators, practical experiences, and position statements without the constraints of a formal peer-review process. The initiative was inspired by the model of the Dagstuhl seminar series²¹, and was launched at the *21st European Conference on Multi-Agent Systems (EUMAS 2024)*²² in Dublin, Ireland. For the first edition, senior researchers from the EMAS community were invited to present their work on toolkits for engineering agents and MAS. After the event, the invited papers were consolidated through a peer-review round and published in a dedicated volume [33].

¹⁹ Throughout the report, we refer to such agents as *LLM agents* or *language agents*.

²⁰ See also the workshop series on EMAS: <https://emas.in.tu-clausthal.de/> (accessed: 14.10.2025).

²¹ <https://dagstuhl.de/> (accessed: 14.10.2025).

²² <https://euramas.github.io/eumas2024/> (accessed: 14.10.2025).

The 2025 edition was hosted by the *22nd European Conference on Multi-Agent Systems (EUMAS 2025)*²³ in Bucharest, Romania. It was organized in a hybrid format, allowing both in-person and online attendance to remove geographical constraints. This second edition focused on recent developments around Generative AI, inviting position statements to surface (among others):

- fundamental models, ideas, and insights from EMAS that could underpin systems of LLM agents;
- missing elements, opportunities, use cases, and overlooked challenges related to EMAS with Generative AI;
- demonstrators and proof-of-concepts that integrate Generative AI with more traditional EMAS.

We received 12 contributions, which were grouped into three topical sessions for presentation at the event. The original position statements are available on the event website.²⁴ This report begins with summaries of the three main sessions. Section 2 presents contributions related to agent programming with LLMs. Section 3 then zooms out to consider multiple engineering dimensions for MAS, specifically tools and communities. Section 4 discusses issues related to interaction, interoperability, and governance. To conclude, Section 5 builds upon the open discussion held at Agent Toolkits 2025 to highlight key directions for future research and a call to action for the EMAS community.

2 Agent Programming with LLMs

The first session grouped contributions on key opportunities and challenges in using LLMs for agent code generation [131], and engineering agents that incorporate LLMs for decision-making and interaction with humans [31, 35, 1].

Winikoff [131] presented “Towards Using LLMs to (Co-)Create Agent(ic) Systems”: He argues that EMAS concepts are relevant to building agentic software, and that using LLMs as coding partners [119] can help reduce the barrier to adopting ideas from EMAS by the agentic development community. He noted that current LLMs can generate code in a range of agent-oriented programming languages, but a number of challenges and research questions remain. Key challenges include guidance for developers, and ensuring reliability and safety. Key research directions include: (1) developing approaches for how EMAS concepts (e.g., goals, plans, norms, commitments, organizations, values) can be used to engineer agentic software; (2) considering how well can LLMs generate agent(ic) code and how well can they function as development partners?; and (3) developing guidance for developers, including on how to best capture requirements for LLMs.

Collier [31] presented “astra-langchain4j: Experiences Combining LLMs and Agent Programming”: He introduces `astra-langchain4j` [32], a library for interacting with LLMs via the ASTRA programming language [34], enabling the

²³ <https://euramas.github.io/eumas2025/> (accessed: 14.10.2025).

²⁴ <https://interactions.ics.unisg.ch/agent-toolkits-2025/> (accessed: 14.10.2025).

incorporation of agentic decision making in a traditional Agent-Oriented Programming (AOP) language. Applying the library to two classic AI challenges, Tic-Tac-Toe and Towerworld, yielded insights that were in line with other analyses [107]: while it is relatively simple to integrate LLM-based decision making into AOP languages, LLMs currently seem to be limited in complex decision making or contextual reasoning where those decisions are related to external environments. Further, LLMs are not trustworthy, in the sense that they are not consistent in their responses.

Dastani and Prasetya [35] presented “Agentic AI, Where Do We Go from Here?”: The recently popularized LLM-based Agentic AI has the potential to be transformational, but heavy reliance on LLMs carries its own risks due to hallucination—and especially so if the technology is deployed *en masse*. The trend can benefit much from integration with the already well-established symbolic-based AI agent techniques and technologies to strengthen these LLM agents’ ability to reason, collaborate among themselves, and improve their robustness against hallucination.

Aguzzi *et al.* [1] presented “LLMs as Agents, LLMs at the Service of Agents, or Agents at the Service of LLMs?”: They reported on the outcomes of a panel entitled “Agents vs. LLMs” organized during the recent annual meeting of the Italian agents and MAS community, the *26th Workshop From Objects to Agents (WOA 2025)*²⁵. While the WOA community completely agreed that LLMs are not agents, the potential for integration was perceived at different degrees. The lack of methodology in engineering Agentic AI systems, and the lack of capability to guarantee some runtime behaviors and properties, emerged as main limitations of Agentic AI. Also based on the panel discussion, Aguzzi *et al.* devised PlanchBDI, a practical proposal to take the best of the new—subsymbolic, opaque, effective, widespread Generative AI—and the old—symbolic, transparent, “for connoisseurs only”—Belief-Desire-Intention (BDI) [13, 96] approaches. PlanchBDI (for **Plan** and **chat BDI**) may be implemented as a BDI extension where both planning in the absence of existing, pre-defined plans [28], and talking to humans [47, 48] are faced by exploiting LLMs, but the engine that drives the logics of the agent behavior is a classical BDI one. In the authors’ vision, PlanchBDI represents an example of LLMs at the service of agents.

3 Conceptual Dimensions, Tool Use, and Communities

The second session grouped contributions that expand the scope from agent engineering to other engineering dimensions [79]—specifically, the agent environment [97, 114] and social behavior [81].

Muller *et al.* [79] presented “Using (Traditional) Multi-Agent Principles to Suggest Improvements for Agentic AI Platforms”: They propose using one of the frameworks developed in “traditional” EMAS (namely the Vowel model [108]),

²⁵ WOA 2025 (<https://sites.google.com/view/woa2025>) co-organized by Giovanni Ciatto and Viviana Mascardi together with Enrico Blanzieri and Sara Chinellato.

which describes a MAS through five aspects: Agent(s), Environment, Interaction(s), Organization, and User(s) to analyze the core components of Agentic AI and assess their current level of maturity. First, this approach may help clarify the emerging concepts of “AI agent” (which appears quite similar to the notion of intelligent agent in traditional MAS, consisting in autonomous entities able to sense and act on their environment, but having a LLM as their "brain"), and “Agentic AI” (which defines the research field encompassing such AI agents). Second, their analysis shows that while Agentic AI offers powerful new capabilities in natural language reasoning and planning, it still faces significant limitations in supporting complex, adaptive, and decentralized MAS. In fact, in Agentic AI, the *Environment* is still limited to the Web and a few tools, compared to the richness of possibilities in traditional MAS listed in [102, 87]. Agentic AI’s *Interactions* are limited to task delegation, compared to the tens of speech acts defined by [45, 44] or the expressiveness of social commitments [109]. Their *Organization* is relatively static, imposed, and purely hierarchical, compared to the richness of models like MOISE+, which defines structural, functional, normative, and dynamic dimensions [87]. Regarding the User(s)/Developer(s) point of view, Agentic AI streamlines interactions (mostly User → Agent) and facilitates code generation (Developer) via natural language interactions and may be easier to develop than traditional MAS. Finally, using the same Vowel model, the authors developed a methodology for analyzing and comparing the functionalities of Agentic AI platforms. They applied this approach to several leading platforms to help developers identify, with greater ease and precision, the most suitable option for their specific use cases.

Ricci and Ciorrea [97] presented “Empowering the Concept of Tool in Agentic AI: Co-evolving AI Agents and Environments with A&A and Hypermedia MAS”: The ability to use tools is an essential feature of LLM agents. In current practice, however, tools are typically limited to external functions invoked through synchronous Application Programming Interface (API) calls. In contrast, the EMAS literature offers richer perspectives, where the agents’ environment as a whole becomes a first-class design and programming abstraction. Notably, the Agents and Artifacts (A&A) meta-model [98] introduces the *artifact* abstraction to explicitly model tools and resources that agents can dynamically build/discover/share/use to support their individual or collective activities. The artifact abstraction is richer than simple function calls: it includes observable properties and events, an asynchronous model for operations, and manuals. By enabling agents to instrument their environments with artifacts, agents and their environments can then co-evolve in response to increasingly complex activities. Recent work on Hypermedia MAS (hMAS) [29] further supports this co-evolution by allowing agents to use hypermedia for discovering and exploiting their evolving environments in an open-ended manner. Additionally, the proficiency of LLMs for code generation now enables agents not only to instantiate pre-programmed artifact templates but also to generate and extend artifact templates as needed. Ricci and Ciorrea reviewed the use of tools in Agentic AI through the lenses of A&A and hMAS, envisioning (1) AI agents capable of exploiting artifacts, as empowered

tools; (2) systems of heterogeneous agents—either based on Generative AI or more classic cognitive architectures—that work together by exploiting artifact-based environments; and (3) a conceptual foundation to support open-ended discovery and self-improvement of co-evolving agents and environments.

Sorici and Olaru [114] presented “Experience-based Agent Communities in support of Smart Environment Hypermedia MAS”: Their contribution is an instance of applied research tackling a long-standing challenge in Ambient Intelligence (AmI)—translating high-level, often vague user intentions into actionable tasks across heterogeneous devices and services [40]. The position statement introduces a framework that uses hMAS and advances in LLMs [113, 63] to improve how smart environments (e.g., homes, offices, factories) translate human intentions into coordinated device actions. It introduces three interacting agent roles—*UserAssistant*, *EnvExplorer*, and *InteractionSolver*—designed to interpret user goals, manage contextual knowledge, and plan actions collaboratively. A key idea is that of using the “signification” [120] mechanism to build *structured records* that capture the context and rationale behind successful interactions. These are represented within extended W3C Web of Things (WoT) Thing Descriptions [123], allowing agents to store and reason from prior *experiences* and reuse effective action sequences across similar contexts. Furthermore, the contribution provides a key idea for how agent roles can address the *cold start* problem in creating plans for totally new requests: *experience-based agent communities*. The distributed model replaces centralized plan repositories, enables autonomy, selective knowledge (signifiers or plans) exchange, and improved generalization in similar smart environments such as buildings or workplaces.

Nallur [81] presented “Social Simulation and Multi-Agent Systems: A False Dichotomy in the Age of Generative Agents”: His core position is that the long-standing division between Social Simulation and MAS is artificial and counterproductive in the context of modern Generative AI. Both fields, he argues, work within socio-technical systems yet emphasize different aspects: MAS focuses on the *mechanisms of society-building* (roles, goals [69], strategies [82, 112, 83], commitments), while Social Simulation emphasizes *observed collective behavior* and *emergent outcomes* [88] arising from interacting agents. These are complementary problem areas, and the two fields should learn from each other. More specifically, Social Simulation needs more realistic intelligent agents with structured knowledge, ontologies, and negotiation processes (strengths of traditional MAS), while MAS need richer case studies that incorporate complex feedback loops and validated social behaviors (strengths of Social Simulation). He contends that Generative AI, though promising, offers only superficial versions of these capabilities without sufficient transparency or depth. His stance is not pessimistic, but integrative: he envisions combining insights from both fields to create AI-enabled socio-technical systems capable of realistic interaction, learning, and behavioral adaptation. However, achieving this synthesis requires advancing agent validation and modeling techniques beyond their current limits to truly support complex, human-like AI collaborations.

4 Interaction, Interoperability, and Governance

The third session grouped contributions in multi-agent interaction [10], interoperability and coordination [53, 68], and enabling transparency for governing agent behavior [64].

Baldoni *et al.* [10] presented “Multiagent Interaction Models for Agentic AI”: They advocate applying principled, formal multi-agent interaction models toward engineering Agentic AI systems. In particular, they emphasize leveraging the newer, declarative, information-based representations of *protocols* [110] and *social commitments* [23] for engineering flexible, asynchronous, and reliable LLM-driven (“Agentic AI”) agents. Baldoni *et al.* argue that such models may be the key to overcoming the unreliability of current Agentic AI and transforming Agentic AI from mostly wishful thinking to an engineering discipline. They point out that the idea of natural language communications between agents—a current fad—is unlikely to ever take off because of the ambiguity and inefficiency of natural language [24]. Baldoni *et al.* also reiterate the crippling, well-known limitations [109, 22, 27] of outdated multi-agent standards such as KQML [44] and FIPA ACL [45]. The newer information-based representations highlighted above address their limitations, present an opportunity for standardization, and represent a crucial way for the EMAS community to remain relevant to recent developments in AI [24].

Gürçan [53] presented “Coral Protocol: Applying Agent-Oriented Software Engineering Principles to Generative-AI Ecosystems”: He argues that LLM agents excel at single-task autonomy but often struggle with effective *collaboration* [124]. Recent industry responses, such as Google’s A2A [117] and Anthropic’s MCP [4] (see also [41]), have standardised message envelopes but have overlooked the need for higher-level coordination mechanisms. In contrast, research in Agent-Oriented Software Engineering (AOSE) has consistently prioritised collaboration, organisational structures, and protocol design as fundamental concerns [115]. Revisiting and adapting these AOSE insights to contemporary LLM agents is, therefore, both timely and essential. Key AOSE concepts that offer immediate value to Generative-AI-based agents include: explicit roles and organisations [17], interaction protocols [90] and environment modelling [127, 99]. While AOSE principles align naturally with Generative AI systems, emerging challenges include: (i) cryptographic trust and identity management for APIs; (ii) lifecycle engineering for prompt-driven agent updates; (iii) blockchain-based economic alignment; and (iv) emergent debugging tools for tracing cross-agent interactions. In this context, the Coral Protocol [50] demonstrates a pragmatic synthesis illustrating how AOSE guidance enables secure team formation, economic incentives, and tool interoperability at Internet scale.

Lemée and Mayer [68] presented “Language Agents May Enable Interoperability in Heterogeneous Multi-Agent Systems”: They argue that language agents might not only form functional components of MAS, but that they may in addition enable interoperability between cognitively heterogeneous agents [84]—for

instance, between a reinforcement learning (RL) agent and a PRS²⁶-based agent: while the former captures its procedural knowledge in terms of policies (and associated expected rewards), the latter keeps it as plans in a plan library. Language agents might adapt and translate this knowledge across agents (artificial as well as natural ones). Next to such translation, the ability of LLMs to generate tools (potentially including artifacts) at run time [18] is another way they might enable interoperability between agents—for example, by creating personalized tools that can be better used by the targeted agent. Clearly, such translation or tool creation might be relatively expensive—sometimes even prohibitively so. The case the proposal is targeting is one where an individual agent is not able to solve a task (or only at high resource expense, or low success probability), but where translation for another type of agent provides a way out. Achieving this vision requires solving certain challenges: first, the knowledge transmitted by an LLM agent might be inaccurate even if the knowledge provided by the originator agent (e.g., a RL agent) is accurate, because LLMs are not fully reliable. One way they consider to address this issue is to validate the knowledge provided by the LLM agent against the affordances available in the environment. This approach matches what SayCan [15] is doing to derive actionable plans using LLMs. The second challenge is that an LLM agent may not know which protocol (e.g., protocols using KQML [44] or negotiated through Agora [74]) to use to transfer its knowledge to another agent. They consider that the concept of agent profiles [120] could be used by agents to indicate to other agents, such as LLM agents, the protocols they are able to use.

Kampik [64] presented “Why We Need Event Data Models for Agents and Multi-Agent Systems”: He argues that in order to show that agents behave in a desirable manner, organizations need to agree on a meta-model of agent event data. Such data model can then serve as a blueprint of how MAS generate event data, or provide the target representation of so-called extract-transform-load pipelines processing MAS data. Ultimately, this is a prerequisite for executing queries on the traces of agents in MAS that formalize behavioral desiderata. The data model should cover both *(i)* agent (inter)actions and *(ii)* agent internals, thus allowing for explanations and justifications. For *(i)*, a first proposal exists [105]; further refinements can draw from a wealth of research of several communities, notably of Normative MAS [21, 65] and process management [39]. For *(ii)*, agreement is necessary on how agent reasoning and decision-making should be represented. Ideally, efforts can lead to agent event data standards, which could use synergies with ongoing agent standardization efforts, thus facilitating unified agent governance across organizations. A prerequisite of successful meta-model development and standardization efforts is the grounding in real-world use cases and the buy-in of industry stakeholders.

²⁶ Procedural Reasoning System [49]

5 The Road Ahead

During the Agent Toolkits 2025 community session, the position statements presented above informed an open discussion of the opportunities and challenges of integrating EMAS and Generative AI research (especially Agentic AI). In what follows, we synthesize this discussion to identify a few key directions for future research and outline a call to action for the EMAS community.

We group our recommendations for future research into two parts. In Section 5.1, we consider what EMAS has to offer that could benefit Agentic AI. Then, in Section 5.2, we consider how EMAS could benefit from Agentic AI. However, in order to realize these various benefits, a key challenge that goes beyond research is to build conceptual, technological, and social bridges between the EMAS and Agentic AI communities. We consider the broader challenge of bridging these communities in Section 5.3, including the particular roles that can be played by standardisation and teaching.

5.1 How EMAS can benefit Agentic AI

A key feature of the recent developments in Agentic AI is the use of LLMs for decision-making, particularly in tasks requiring reasoning²⁷ and tool use. An LLM agent relies on an LLM that operates in a feedback loop with its external environment [116]: it acts on the environment by invoking a tool, typically understood as a synchronous function call [125], and may receive a corresponding observation in return. This simple machinery has proven effective for a variety of tasks, but LLMs also suffer from several fundamental limitations [77, 116, 125]: they generate non-factual but plausible responses (referred to as *hallucinations* or *bullshit* [57]), struggle with tasks requiring complex reasoning, and lack transparency. These limitations are problematic for agents that have to plan a course of action or coordinate with one another to achieve a goal—and are especially troublesome when considering the scale that Agentic AI might achieve. Furthermore, there is growing concern about the high—and sometimes prohibitive—costs of training and using LLMs, both financially²⁸ and environmentally²⁹ [62, 89, 38]. The Agent Toolkits 2025 position statements echo these limitations and lay out a question for the broader EMAS community:

How can we engineer autonomous agents that use LLMs, but leverage EMAS concepts and techniques to improve their effectiveness, flexibility, reliability, and transparency?

²⁷ Note that *LLM reasoning* is reminiscent of but different from reasoning as understood in symbolic AI. See [77] for a more detailed discussion.

²⁸ In a recent engineering article, Anthropic estimates that an LLM agent uses 4x more tokens than chat interactions, and an LLM-based MAS uses 15x more tokens [6].

²⁹ Relatively simple text summarization tasks by an LLM generate an estimated 8 *gCO₂eq* [72], compared to around 0.003 *gCO₂eq* for a regular Web search that might yield similar value (factor ~3,000); for image generation tasks, the factor is ~150,000.

EMAS research provides a rich body of work on theories, architectures, methodologies, paradigms, and languages for designing and implementing autonomous agents. It is therefore only natural to ask how these results might help us rethink how to structure, design, and build new *hybrid* agent architectures that mitigate the current weaknesses of LLM agents. Initial explorations in this direction include [60, 47, 28, 32]. There is interest in the Agentic AI community in borrowing abstractions and lessons learned from other areas of AI. One example is the work on cognitive architectures for language agents by Sumers *et al.* [116], which draws on key ideas promoted by Allen Newell through Soar [66] and, more broadly, knowledge-based agents [85].

Another research opportunity is the emergence of *context engineering* in Agentic AI [5], which is reminiscent of the *environment as a first-class abstraction* in EMAS research [127]. Tool use is already a fundamental feature of LLM agents—and designing tools that agents can use effectively is becoming increasingly important [7]. The shift from *prompt engineering* to *context engineering* brings greater focus on “the set of strategies for curating and maintaining the optimal set of tokens (information) during LLM inference, including all the other information that may land there outside of the prompts” [5]. This additional information includes the set of tools available to agents, the descriptions of those tools, and any other external resources relevant to their activities. EMAS research provides models and technologies for designing and implementing environments that offer various levels of support to agents—from active perception [128] to shaping their environment so that it better meets their needs [98]. Recent work on hypermedia-based agent environments [29] brings these ideas to the open Web, in line with mechanisms for run-time tool discovery and usage in initiatives such as MCP [4], the Universal Tool Calling Protocol (UTCP)³⁰, or the Eclipse Language Model Operating System (LMOS)³¹. Integrating these lines of research could open the door to a new generation of LLM agents that are aware of their environment and can reflect on it to operate more effectively. This is highly relevant to Agentic AI research, as the community discovers that context and its management play a pivotal role in the efficiency of LLM agents.

Beyond individual agents and their environments, EMAS research provides models of interaction—including protocols and normative abstractions—that are indispensable world models: they tell an agent the possible actions it may perform and the normative consequences of doing so. Moreover, these interaction models offer a natural opportunity to combine LLM-based and symbolic approaches. The advantages of declarative information-based representations of interactions for agent programming have been demonstrated via several implemented programming models, including for plain Python and BDI agents [26, 25, 11, 23, 27]. Investigating architectures for LLM agents that leverage such interaction models is, therefore, an obvious and promising research direction. The potential for such a synthesis should motivate the EMAS community to explore declarative, higher-level models of interaction and associated programming mod-

³⁰ <https://www.utcp.io/> (accessed: 31.10.2025).

³¹ <https://eclipse.dev/lmos/> (accessed: 31.10.2025).

els (see [111] for an example of such an effort)—ultimately, to be used in environments as open as the Web. The objective is then to aspire to *declarative specifications of interaction protocols that are so simple that even an LLM can reliably work with them*.

The lack of transparency common to many AI technologies, including LLMs, has motivated substantial research on how to provide means of engineering autonomous systems that can explain their actions, ideally in human-meaningful terms [100, 101, 78, 132]. Although there is a whole body of work on Explainable AI (XAI), much of it is concerned with explaining machine learning (termed “data-driven XAI” by Anjomshoae *et al.* [2]), with much less work on explaining autonomous agents (termed “goal-driven XAI” [2], or “explainable agency” [67]). Miller [78] argued that such work should take into account prior work in the social sciences. Subsequently, Winikoff *et al.* [132] highlighted a natural correspondence between the BDI model and the concepts that humans use to explain their actions [73]. This correspondence means that techniques for explaining BDI agents (e.g. [54, 14, 55, 132]) naturally provide explanations in human-meaningful terms—and offer a promising opportunity for research on explainable behavior in LLM agents.

However, while transparency can help with uncovering system deficiencies, it is clearly important to try and avoid significant deficiencies altogether. This has motivated work on aligning LLMs with human values [126, 56, 8, 86]. Common approaches include Reinforcement Learning from Human Feedback (RLHF) [52, 9], Direct Preference Optimization (DPO) [94], and more recent work on *explicit* moral alignment goals [118]. However, with “alignment” of LLMs, we cannot circumvent their very nature *in principle* [104] (not peer-reviewed). Hence, normative abstractions and mechanisms from EMAS research may provide new avenues for explicitly defining, monitoring, and governing standards of behavior in a new generation of norm-aware LLM agents (see also [20]). Furthermore, such abstractions and mechanisms may even be required for provable norm-compliant behavior, and could also support LLM agents in generating accounts of their behavior. A prerequisite for governance and accountability is then agreeing on how traces of agent behavior and their interactions should be represented—for example, in the form of an agent event data (meta-)model [64]. Given sufficient interest from industry and practitioners, such meta-models could mature into open standards to enable transparency across systems.

Finally, we warn that the consequences of LLMs’ unreliability might be underestimated in large-scale Agentic AI systems, where unreliability could be strategically exploited while incurring little cost to the attacker, and we point out that EMAS research provides considerable work on formal verification that may be leveraged to ensure reliability [37, 43, 46].

5.2 How Generative AI can benefit EMAS

The previous section emphasizes the rich body of work on engineering autonomous agents and MAS. However, despite a number of successful applications (e.g., see [80]), the adoption of classical agent technologies in industry has

remained limited [71]—especially when contrasted with the attention given to Agentic AI in recent years. One contributing factor may be the high entry barrier associated with learning agent-oriented programming—whereas the appeal of LLM agents lies in their promise to tackle a wide range of tasks out of the box with little prior programming [59]. Importantly, the entry barrier to building LLM-based applications has also been significantly lowered through frameworks that support the integration of LLMs, such as LangChain³² and its associated tooling. The (careful) adoption and integration of Generative AI—and especially LLMs—into classical agent systems could help lower the entry barrier for developers and offer a range of new opportunities. The second question we raise to the broader EMAS community, hence, is:

How can we leverage Generative AI to lower the entry barrier to engineering autonomous agents, and to improve the agents’ adaptiveness, interoperability, and usability?

One promising opportunity that Generative AI brings for lowering the entry barrier to agent-oriented programming is LLM-assisted code generation [130]. Agent programming languages build on high-level, intuitive abstractions such as beliefs, goals, and plans [106]. This view has a rich theoretical foundation tracing back to Daniel Dennett’s intentional stance [36]—in which systems are ascribed mental states such as beliefs, desires, and intentions—and John McCarthy’s similar view on ascribing mental qualities to machines [75]. In both views, the focus is not on the legitimacy of these ascriptions but on their usefulness for understanding the structure and behavior of complex systems. EMAS research explored the use of such high-level abstractions to promote low-code development (e.g., [30]) and visual programming languages (e.g., [61, 16]) for agent systems. LLM-assisted code generation could take these ideas a step further, potentially bringing agent-oriented programming to tech-savvy users without a formal programming background.

Going further, a drawback of classical knowledge-based agents is that they rely on hand-crafted knowledge and rules, such as beliefs and plans in BDI agents [49, 95], which do not generalize well to new or highly dynamic environments [116]. Approaches explored in EMAS research for decoupling agents from their environments include automated planning [76], defining a standard environment interface [12], and using semantic hypermedia to abstract away from the agent’s deployment context [29]. While these approaches help mitigate the issue, they still rely on manual (knowledge) engineering efforts [19]. Other approaches integrate BDI reasoning with learning methods [42], specifically reinforcement learning, but with additional training costs. In contrast, the appeal of LLMs is that their pre-training on a very large, diverse corpus allows them to learn an effective distribution over string completions, which gives them the remarkable ability to solve a wide range of tasks out of the box [116] (note here also the increasing research focus on large multimodal models). It is then only natural to ask how these capabilities could help with engineering more flexible

³² See <https://www.langchain.com/> (accessed: 06.11.2025)

knowledge-based agents, where the intrinsic knowledge of LLMs could complement the knowledge represented explicitly within an agent’s belief base or plan library (e.g., see [28,32]). Although current LLMs are known not to be very good at planning in the classical sense [122], they may perform better if asked to generate BDI-style plans—or simply to match abstract actions from pre-defined plans to affordances discovered at run-time (see also [121]).

Another promising opportunity is the use of Generative AI to support human interaction and to integrate heterogeneous types of agents—whether by interpreting and generating natural language or by mediating between heterogeneous representation formats and even interaction modalities. LLM-assisted interaction with humans is an immediate benefit, given the LLMs’ ability to generate high-quality natural language across a range of languages. Early explorations in this direction include the work of Ichida *et al.* [60], which enables developers to instruct BDI agents using natural language as the programming interface, and the work of Gatti *et al.* [47], which focuses on interaction with end users. Furthermore, LLMs may help integrate heterogeneous types of agents (e.g., LLM agents, BDI agents, RL agents) in open environments such as the Web [68]. For example, while a classical BDI agent cannot directly process the natural-language description of an MCP tool, an LLM-based translation or mediation mechanism could facilitate such interoperability. A key challenge for the EMAS community is thus to explore how Generative AI might help integrate classical agents more effectively into systems of LLM agents—and what benefits such integration could bring to the Agentic AI community.

5.3 The need to bridge communities

The previous sections outlined several research opportunities that could benefit from closer collaboration between the EMAS and Agentic AI communities. Engineering autonomous systems is not new—the EMAS community has been advancing this field for decades [58,130]. While LLMs face well-known issues with reliability, transparency, and planning [122,63], research from the EMAS community could help address these shortcomings. At the same time, LLMs—and Generative AI more broadly—unlock new capabilities for autonomous agents and MAS, and thus new research opportunities for the EMAS community. Realizing this potential requires a systematic transfer of knowledge that builds conceptual, technological, and social bridges between the two communities. A key challenge for the EMAS community is thus to make its ongoing and historical work more visible, both to organizations developing Agentic AI technologies and to the broad range of stakeholders who are commenting on or considering adopting these technologies. In doing so, two key messages might be: “LLMs have significant issues with reliability and transparency”, and “the EMAS community, which has been working on engineering autonomous systems for decades, can help”.

Another particular challenge, as noted by Winikoff [129, §5], is that the EMAS community is largely European and university-based. In contrast, Agentic AI development is currently occurring predominantly in the US in large

companies. This asymmetry makes bridging these two communities particularly challenging. One possible approach for the EMAS community is to focus on building bridges not just with Agentic AI technology developers, but also with prospective *adopters* who are more likely to be pragmatic about Agentic AI technologies—and, hence, receptive to the benefits of integrating EMAS technologies. A complementary effort would be to organize regular academic workshops strategically—across continents and co-located with conferences that represent both communities. Prestigious venues such as the Dagstuhl Seminar series³³ could also help attract key researchers and practitioners from around the world. In addition, international standardization bodies can facilitate communication by providing regular meeting venues for the two communities.

One example of such a bridging effort is the *W3C Autonomous Agents on the Web (WebAgents) Community Group (CG)*, which brings together researchers and practitioners across three communities broadly identified as: *Autonomous Agents and MAS*, *Semantic Web and Linked Data*, and *Web Architecture and the Web of Things*. The core of this community was consolidated through four biannual events: the *First International Workshop on Hypermedia Multi-Agent Systems (HyperAgents 2019)*³⁴, co-located with *The Web Conference 2019* in San Francisco, CA, USA; Dagstuhl Seminar 21702 (Feb. 2021)³⁵; Dagstuhl Seminar 23081 (Feb. 2023)³⁶; and the *Second International Workshop on Hypermedia Multi-Agent Systems (HyperAgents 2025)*³⁷, co-located with the *European Conference on Artificial Intelligence (ECAI 2025)* in Bologna, Italy. The W3C WebAgents CG was created following the second Dagstuhl seminar and, at the time of writing, includes 150 participants across five continents. Many of the active participants are part of (or close to) the EMAS community—and, more recently, the CG has increasingly focused on Agentic AI, attracting a growing number of participants from that community as well.

Standardization and *teaching* are two particularly important avenues for bridging communities and facilitating transfer into practice. Standardization, when done well, can be a powerful mechanism for raising awareness and promoting the adoption of key ideas and approaches established in the research community—and, equally important, for sharing lessons learned from past missteps. Another critical task that standardization activities can help with is establishing a shared terminology that facilitates knowledge transfer and joint work between the two communities. Teaching is essential to ensure that the next generation of professionals is equipped with the insights and skills needed to design autonomous agents and MAS that build on the latest developments from both EMAS and Agentic AI—rather than seeing them as two distinct and disconnected worlds. This would benefit from a systematic revision of teaching activities at all levels to integrate ideas from both communities into a coher-

³³ <https://www.dagstuhl.de/en/seminars/dagstuhl-seminars> (accessed: 04.11.2025).

³⁴ <https://www2019.hyperagents.org> (accessed: 04.11.2025).

³⁵ <https://dagstuhl.de/21072> (accessed: 04.11.2025).

³⁶ <https://dagstuhl.de/23081> (accessed: 04.11.2025).

³⁷ <https://ecai2025.hyperagents.org> (accessed: 04.11.2025).

ent curriculum—but we acknowledge the complexities of such systemic changes. In addition, individual courses could invite students to appropriately integrate relevant concepts from both fields, both conceptually and in implementation projects.

Ghose’s [51] advice on Service-Oriented Computing (SOC), made more than 15 years ago, seems pertinent. He noted that “*An informal analysis of the papers presented at a recent major conference on service-oriented computing suggests that at least 60% of these had agent technology underpinnings in some form or the other*”. As the position statements presented at Agent Toolkits 2025 suggest, something similar may now be happening with Agentic AI. Ghose went on to argue that the EMAS community should do two things: (1) learn from the SOC community how to better achieve industry adoption—particularly noting the importance of defining a core research agenda and offering a simple value proposition that requires incremental rather than radical changes; and (2) engage with the SOC community (both research and industry), bearing in mind the importance of using standard industry terms. As Ghose notes in his closing, terminology matters: Shakespeare was wrong when he wrote that “*that which we call a rose, by any other name would smell as sweet*”.

References

1. Aguzzi, G., Ciatto, G., Ferrando, A., Gatti, A., Mascardi, V.: LLMs as agents, LLMs at the service of agents, or agents at the service of LLMs? (2025), Agent Toolkits 2025. Available at <https://agent-toolkits.github.io/Agent-Toolkits-2025/papers/aguzzi-et-al.pdf>
2. Anjomshoae, S., Najjar, A., Calvaresi, D., Främling, K.: Explainable agents and robots: Results from a systematic literature review. In: AAMAS. pp. 1078–1088 (2019), <http://dl.acm.org/citation.cfm?id=3331806>
3. Anthropic: Building effective agents. <https://www.anthropic.com/engineering/building-effective-agents> (2024), accessed: 14.10.2025
4. Anthropic: Model Context Protocol (MCP). <https://modelcontextprotocol.io/> (2024), accessed: 28.04.2025
5. Anthropic: Effective context engineering for ai agents. <https://www.anthropic.com/engineering/effective-context-engineering-for-ai-agents> (2025), accessed: 14.10.2025
6. Anthropic: How we built our multi-agent research system. <https://www.anthropic.com/engineering/multi-agent-research-system> (2025), accessed: 14.10.2025
7. Anthropic: Writing effective tools for agents—with agents. <https://www.anthropic.com/engineering/writing-tools-for-agents> (2025), accessed: 14.10.2025
8. Anwar, U., Saporov, A., Rando, J., Paleka, D., Turpin, M., Hase, P., Lubana, E.S., Jenner, E., Casper, S., Sourbut, O., Edelman, B.L., Zhang, Z., Günther, M., Korinek, A., Hernandez-Orallo, J., Hammond, L., Bigelow, E., Pan, A., Langosco, L., Korbak, T., Zhang, H., Zhong, R., hEigeartaigh, S.O., Recchia, G., Corsi, G., Chan, A., Anderljung, M., Edwards, L., Petrov, A., de Witt, C.S., Motwan, S.R., Bengio, Y., Chen, D., Torr, P.H.S., Albanie, S., Maharaj, T.,

- Foerster, J., Tramer, F., He, H., Kasirzadeh, A., Choi, Y., Krueger, D.: Foundational challenges in assuring alignment and safety of large language models (2024), <https://arxiv.org/abs/2404.09932>
9. Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., Joseph, N., Kadavath, S., Kernion, J., Conerly, T., El-Showk, S., Elhage, N., Hatfield-Dodds, Z., Hernandez, D., Hume, T., Johnston, S., Kravec, S., Lovitt, L., Nanda, N., Olsson, C., Amodei, D., Brown, T., Clark, J., McCandlish, S., Olah, C., Mann, B., Kaplan, J.: Training a helpful and harmless assistant with reinforcement learning from human feedback (2022), <https://arxiv.org/abs/2204.05862>
 10. Baldoni, M., Baroglio, C., Chopra, A.K., Marengo, E., Micalizio, R., Tedeschi, S.: Multiagent interaction models for agentic AI (2025), Agent Toolkits 2025. Available at <https://agent-toolkits.github.io/Agent-Toolkits-2025/papers/baldoni-et-al.pdf>
 11. Baldoni, M., Christie, S.H., Singh, M.P., Chopra, A.K.: Orpheus: Engineering multiagent systems via communicating agents. In: Proceedings of the 39th AAAI Conference on Artificial Intelligence (AAAI). pp. 23135–23143. AAAI, Philadelphia (Feb 2025). <https://doi.org/10.1609/aaai.v39i22.34478>
 12. Behrens, T.M., Hindriks, K.V., Dix, J.: Towards an environment interface standard for agent platforms. *Annals of Mathematics and Artificial Intelligence* **61**(4), 261–295 (Apr 2011). <https://doi.org/10.1007/s10472-010-9215-9>, <https://doi.org/10.1007/s10472-010-9215-9>
 13. Bratman, M.: *Intention, Plans, and Practical Reason*. Cambridge, MA: Harvard University Press, Cambridge (1987)
 14. Broekens, J., Harbers, M., Hindriks, K.V., van den Bosch, K., Jonker, C.M., Meyer, J.C.: Do you get it? User-evaluated explainable BDI agents. In: *Multiagent System Technologies (MATES)*. LNCS, vol. 6251, pp. 28–39. Springer (2010). https://doi.org/10.1007/978-3-642-16178-0_5
 15. Brohan, A., Chebotar, Y., Finn, C., Hausman, K., Herzog, A., Ho, D., Ibarz, J., Irpan, A., Jang, E., Julian, R., et al.: Do as I can, not as I say: Grounding language in robotic affordances. In: *Conference on robot learning*. pp. 287–318. PMLR (2023)
 16. Burattini, S., Ricci, A., Mayer, S., Vachtsevanou, D., Lemee, J., Ciortea, A., Croatti, A.: Agent-oriented visual programming for the web of things (2025), <https://arxiv.org/abs/2511.13158>
 17. Cabri, G., Ferrari, L., Zambonelli, F.: Role-based approaches for engineering interactions in large-scale multi-agent systems. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **2940**, 243 – 263 (2004). https://doi.org/10.1007/978-3-540-24625-1_14
 18. Cai, T., Wang, X., Ma, T., Chen, X., Zhou, D.: Large language models as tool makers. In: *The Twelfth International Conference on Learning Representations (2024)*, <https://openreview.net/forum?id=qV83K9d5WB>
 19. Calbimonte, J.P., Ciortea, A., Kampik, T., Mayer, S., Payne, T.R., Tamma, V., Zimmermann, A.: Autonomy in the age of knowledge graphs : vision and challenges. *Transactions on Graph Data and Knowledge (TGDK) (ARTICLE)*, 22 p. <https://doi.org/https://doi.org/10.4230/tgdk.1.1.13>, <http://arodes.heso.ch/record/13904>
 20. Calvanese, D., De Giacomo, G., Kampik, T., Lesperance, Y., Marrella, A., Matta, A.: Autonomy in business process execution: Why we need first-class abstractions

- for goals and normative frames. In: PMAI'25: 3rd International Workshop on Process Management in the AI era (2025), <https://tinyurl.com/PMAI2025Calvanese>
21. Chopra, A., van der Torre, L., Verhagen, H.: Handbook of Normative Multiagent Systems. College Publications (2018)
 22. Chopra, A.K., Artikis, A., Bentahar, J., Colombetti, M., Dignum, F., Fornara, N., Jones, A.J.I., Singh, M.P., Yolum, P.: Research directions in agent communication. *ACM Transactions on Intelligent Systems and Technologies* **4**(2), 20:1–20:23 (2013)
 23. Chopra, A.K., Baldoni, M., Christie, S.H., Singh, M.P.: Azorus: Commitments over protocols for BDI agents. In: Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2025, Detroit, MI, USA, May 19-23, 2025. pp. 490–499. International Foundation for Autonomous Agents and Multiagent Systems (2025). <https://doi.org/10.5555/3709347.3743564>
 24. Chopra, A.K., Christie, S.H., Singh, M.P.: Tools for implementing multiagent systems based on protocols. In: Agent Toolkits. Springer (2025), in press
 25. Christie, S.H., Singh, M.P., Chopra, A.K.: Kiko: Programming agents to enact interaction protocols. In: Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2023, London, United Kingdom, 29 May 2023 - 2 June 2023. pp. 1154–1163. IFAAMAS, London (May 2023). <https://doi.org/10.5555/3545946.3598758>
 26. Christie, S.H., Chopra, A.K., Singh, M.P.: Mandrake: Multiagent systems as a basis for programming fault-tolerant decentralized applications. *Auton. Agents Multi Agent Syst.* **36**(1), 16:1–16:30 (Apr 2022). <https://doi.org/10.1007/s10458-021-09540-8>
 27. Christie, S.H., Singh, M.P., Chopra, A.K.: Argus: Programming with communication protocols in a belief-desire-intention architecture. *Artificial Intelligence* **348**, 104398:1–104398:35 (Aug 2025). <https://doi.org/10.1016/j.artint.2025.104398>
 28. Ciatto, G., Aguzzi, G., Battistini, R., Baiardi, M., Burattini, S., Ricci, A.: Exploiting GenAI for Plan Generation in BDI Agents. In: European Conference on Artificial Intelligence. Bologna (2025)
 29. Ciortea, A., Castellucci, M., Schultz, K., Lemée, J., Vachtsevanou, D., Gandon, F., Mayer, S., Berger, V., Burattini, S., Ricci, A., Boissier, O.: Yggdrasil: An artifact-based framework for hypermedia multi-agent systems. In: Collier, R., Mascardi, V., Ricci, A. (eds.) *Agents and Multi-Agent Systems Development – Platforms, Toolkits, Technologies*. Springer (2025)
 30. Ciortea, A., Mayer, S., Michahelles, F.: Repurposing manufacturing lines on the fly with multi-agent systems for the web of things. In: Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems. p. 813–822. AAMAS '18, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2018)
 31. Collier, R.: astra-langchain4j: Experiences combining LLMs and agent programming (2025), *Agent Toolkits 2025*. Available at <https://agent-toolkits.github.io/Agent-Toolkits-2025/papers/collier.pdf>
 32. Collier, R., Beaumont, K., Ciortea, A.: astra-langchain4j: Experiences Combining LLMs and Agent Programming. In: 22nd European Conference on Multi-Agent Systems (EUMAS 2025) (2025)
 33. Collier, R., Mascardi, V., Ricci, A. (eds.): *Agents and Multi-Agent Systems Development – Platforms, Toolkits, Technologies*. Springer-Verlag GmbH (2025). <https://doi.org/https://doi.org/10.1007/978-3-032-01082-7>

34. Collier, R.W., Russell, S., Lillis, D.: Reflecting on agent programming with AgentSpeak(L). In: International Conference on Principles and Practice of Multi-Agent Systems. pp. 351–366. Springer (2015)
35. Dastani, M., Prasetya, I.: Agentic AI, where do we go from here? (2025), Agent Toolkits 2025. Available at <https://agent-toolkits.github.io/Agent-Toolkits-2025/papers/dastani-prasetya.pdf>
36. Dennett, D.C.: The Intentional Stance. MIT press (1987)
37. Dennis, L.A., Fisher, M.: Verifiable Autonomous Systems: Using Rational Agents to Provide Assurance about Decisions Made by Machines. Cambridge University Press, Cambridge, United Kingdom ; New York, NY, USA (2023)
38. Desislavov, R., Martínez-Plumed, F., Hernández-Orallo, J.: Trends in ai inference energy consumption: Beyond the performance-vs-parameter laws of deep learning. *Sustainable Computing: Informatics and Systems* **38**, 100857 (2023). <https://doi.org/https://doi.org/10.1016/j.suscom.2023.100857>, <https://www.sciencedirect.com/science/article/pii/S2210537923000124>
39. Diba, K., Batoulis, K., Weidlich, M., Weske, M.: Extraction, correlation, and abstraction of event data for process mining. *WIREs Data Mining and Knowledge Discovery* **10**(3), e1346 (2020). <https://doi.org/https://doi.org/10.1002/widm.1346>
40. Ducatel, K., Union européenne. Technologies de la société de l’information, Union européenne. Institut d’études de perspectives technologiques, Union européenne. Société de l’information conviviale: Scenarios for ambient intelligence in 2010. <https://op.europa.eu/en/publication-detail/-/publication/42f2b572-64c5-4ef3-b9af-2d44de887a254> (2001)
41. Ehtesham, A., Singh, A., Gupta, G.K., Kumar, S.: A survey of agent interoperability protocols: Model context protocol (MCP), agent communication protocol (ACP), agent-to-agent protocol (A2A), and agent network protocol (ANP) (2025), <https://arxiv.org/abs/2505.02279>
42. Erduran, Ö.I.: Machine learning for cognitive bdi agents: a compact survey. In: Proceedings of the 15th International Conference on Agents and Artificial Intelligence. Volume 1: ICAART. vol. 1.2023, pp. 257 – 268 (2023). <https://doi.org/10.5220/0011678100003393>
43. Ferrando, A., Dennis, L.A., Cardoso, R.C., Fisher, M., Ancona, D., Mascardi, V.: Toward a Holistic Approach to Verification and Validation of Autonomous Cognitive Systems. *ACM Trans. Softw. Eng. Methodol.* **30**(4), 43:1–43:43 (2021). <https://doi.org/10.1145/3447246>
44. Finin, T., Fritzson, R., McKay, D., McEntire, R.: KQML as an agent communication language. In: Proceedings of the Third International Conference on Information and Knowledge Management. pp. 456–463. CIKM ’94, Association for Computing Machinery, New York, NY, USA (1994). <https://doi.org/10.1145/191246.191322>
45. FIPA: FIPA ACL Message Structure Specification. Standard, Foundation for Intelligent Physical Agents (2002), <http://www.fipa.org/specs/fipa00061/SC00061G.html>, accessed: July 16th, 2025
46. Fisher, M., Mascardi, V., Rozier, K.Y., Schlingloff, B.H., Winikoff, M., Yorke-Smith, N.: Towards a framework for certification of reliable autonomous systems. *Auton. Agents Multi Agent Syst.* **35**(1), 8 (2021). <https://doi.org/10.1007/S10458-020-09487-2>

47. Gatti, A., Mascardi, V., Ferrando, A.: ChatBDI: Think BDI, Talk LLM. In: Das, S., Nowé, A., Vorobeychik, Y. (eds.) Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2025, Detroit, MI, USA, May 19-23, 2025. pp. 2541–2543. International Foundation for Autonomous Agents and Multiagent Systems / ACM (2025). <https://doi.org/10.5555/3709347.3743930>
48. Gatti, A., Mascardi, V., Ferrando, A.: Let me talk to you! Natural language interaction between humans and BDI agents via ChatBDI. In: European Conference on Artificial Intelligence. Bologna (2025)
49. Georgeff, M.P., Lansky, A.L.: Reactive Reasoning and Planning. In: Forbus, K.D., Shrobe, H.E. (eds.) Proceedings of the 6th National Conference on Artificial Intelligence. Seattle, WA, USA, July 1987. pp. 677–682. Morgan Kaufmann (1987), <http://www.aaai.org/Library/AAAI/1987/aaai87-121.php>
50. Georgio, R.J., Forder, C., Deb, S., Rahimov, A., Carroll, P., Önder Gürcan: Coral protocol: Open infrastructure connecting the internet of agents (2025), <https://arxiv.org/abs/2505.00749>
51. Ghose, A.: Industry traction for MAS technology: would a rose by any other name smell as sweet? *Int. J. Agent Oriented Softw. Eng.* **3**(4), 397–401 (2009). <https://doi.org/10.1504/IJAOSE.2009.025318>
52. Glaese, A., McAleese, N., Trębacz, M., Aslanides, J., Firoiu, V., Ewalds, T., Rauh, M., Weidinger, L., Chadwick, M., Thacker, P., Campbell-Gillingham, L., Uesato, J., Huang, P.S., Comanescu, R., Yang, F., See, A., Dathathri, S., Greig, R., Chen, C., Fritz, D., Elias, J.S., Green, R., Mokra, S., Fernando, N., Wu, B., Foley, R., Young, S., Gabriel, I., Isaac, W., Mellor, J., Hassabis, D., Kavukcuoglu, K., Hendricks, L.A., Irving, G.: Improving alignment of dialogue agents via targeted human judgements (2022), <https://arxiv.org/abs/2209.14375>
53. Önder Gürcan: Coral protocol: Applying agent-oriented software engineering principles to generative-ai ecosystems (2025), Agent Toolkits 2025. Available at <https://agent-toolkits.github.io/Agent-Toolkits-2025/papers/gurcan.pdf>
54. Harbers, M.: Explaining Agent Behavior in Virtual Training. SIKS dissertation series no. 2011-35, SIKS (Dutch Research School for Information and Knowledge Systems) (2011)
55. Harbers, M., van den Bosch, K., Meyer, J.C.: Design and evaluation of explainable BDI agents. In: Intelligent Agent Technology (IAT). pp. 125–132. IEEE (2010). <https://doi.org/10.1109/WI-IAT.2010.115>
56. Hendrycks, D., Burns, C., Basart, S., Critch, A., Li, J., Song, D., Steinhardt, J.: Aligning ai with shared human values (2023), <https://arxiv.org/abs/2008.02275>
57. Hicks, M.T., Humphries, J., Slater, J.: Chatgpt is bullshit. *Ethics and Information Technology* **26**(2), 38 (2024). <https://doi.org/10.1007/s10676-024-09775-5>, <https://doi.org/10.1007/s10676-024-09775-5>
58. Hindriks, K.V.: The Shaping of the Agent-Oriented Mindset - Twenty Years of Engineering MAS. In: Dalpiaz, F., Dix, J., Riemsdijk, M.B.v. (eds.) Engineering Multi-Agent Systems - Second International Workshop, EMAS 2014, Paris, France, May 5-6, 2014, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8758, pp. 1–14. Springer (2014). https://doi.org/10.1007/978-3-319-14484-9_1
59. Huang, W., Abbeel, P., Pathak, D., Mordatch, I.: Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In: Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., Sabato, S. (eds.) Proceedings of the 39th International Conference on Machine Learning. Proceedings of

- Machine Learning Research, vol. 162, pp. 9118–9147. PMLR (17–23 Jul 2022), <https://proceedings.mlr.press/v162/huang22a.html>
60. Ichida, A.Y., Meneguzzi, F., Cardoso, R.C.: Bdi agents in natural language environments. In: Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems. p. 880–888. AAMAS '24, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2024)
 61. Jayatilleke, G.B., Padgham, L., Winikoff, M.: Evaluating a model driven development toolkit for domain experts to modify agent based systems. In: Padgham, L., Zambonelli, F. (eds.) Agent-Oriented Software Engineering VII. pp. 190–207. Springer Berlin Heidelberg, Berlin, Heidelberg (2007)
 62. Jegham, N., Abdelatti, M., Elmoubarki, L., Hendawi, A.: How hungry is ai? benchmarking energy, water, and carbon footprint of llm inference (2025), <https://arxiv.org/abs/2505.09598>
 63. Kambhampati, S., Valmeekam, K., Guan, L., Verma, M., Stechly, K., Bhambri, S., Saldyt, L., Murthy, A.: Position: LLMs Can't Plan, But Can Help Planning in LLM-Modulo Frameworks. In: Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024. OpenReview.net (2024), <https://openreview.net/forum?id=Th8JPEmH4z>
 64. Kampik, T.: Why we need event data models for agents and multi-agent systems (2025), Agent Toolkits 2025. Available at <https://agent-toolkits.github.io/Agent-Toolkits-2025/papers/kampik.pdf>
 65. Kampik, T., Mansour, A., Boissier, O., Kirrane, S., Padget, J.A., Payne, T.R., Singh, M.P., Tamma, V., Zimmermann, A.: Governance of Autonomous Agents on the Web: Challenges and Opportunities. ACM Trans. Internet Techn. **22**(4), 104:1–104:31 (2022). <https://doi.org/10.1145/3507910>
 66. Laird, J.E., Newell, A., Rosenbloom, P.S.: Soar: An architecture for general intelligence. Artificial Intelligence **33**(1), 1–64 (1987). [https://doi.org/https://doi.org/10.1016/0004-3702\(87\)90050-6](https://doi.org/https://doi.org/10.1016/0004-3702(87)90050-6), <https://www.sciencedirect.com/science/article/pii/0004370287900506>
 67. Langley, P., Meadows, B., Sridharan, M., Choi, D.: Explainable agency for intelligent autonomous systems. In: AAAI Conference on Artificial Intelligence. pp. 4762–4764. AAAI Press (2017), <http://aaai.org/ocs/index.php/IAAI/IAAI17/paper/view/15046>
 68. Lemée, J., Mayer, S.: Language agents may enable interoperability in heterogeneous multi-agent systems (2025), Agent Toolkits 2025. Available at <https://agent-toolkits.github.io/Agent-Toolkits-2025/papers/lemee-mayer.pdf>
 69. Lewis, P.R., Goldingay, H., Nallur, V.: It's good to be different: Diversity, heterogeneity, and dynamics in collective systems. In: Self-adaptive and self-organizing systems workshops (SASOW), 2014 IEEE eighth international conference on. pp. 84–89. IEEE (2014)
 70. Li, G., Hammoud, H., Itani, H., Khizbullin, D., Ghanem, B.: Camel: Communicative agents for "mind" exploration of large language model society. In: Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S. (eds.) Advances in Neural Information Processing Systems. vol. 36, pp. 51991–52008. Curran Associates, Inc. (2023)
 71. Logan, B.: An agent programming manifesto. International Journal of Agent-Orientated Software Engineering **6**, 187–210 (2018), <https://nottingham-repository.worktribe.com/output/871572>, 12 month embargo. KJH 31.07.2017. Updated OL 21.08.2018

72. Luccioni, S., Jernite, Y., Strubell, E.: Power hungry processing: Watts driving the cost of ai deployment? In: Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency. p. 85–99. FAccT '24, Association for Computing Machinery, New York, NY, USA (2024). <https://doi.org/10.1145/3630106.3658542>, <https://doi.org/10.1145/3630106.3658542>
73. Malle, B.F.: How the Mind Explains Behavior. MIT Press (2004), ISBN: 9780262134453
74. Marro, S., La Malfa, E., Wright, J., Li, G., Shadbolt, N., Wooldridge, M., Torr, P.: A scalable communication protocol for networks of large language models. arXiv preprint arXiv:2410.11905 (2024)
75. McCarthy, J.: Ascribing mental qualities to machines (1979), <https://apps.dtic.mil/sti/html/tr/ADA071423/>, Technical Report, AIM-326, Stanford University
76. Meneguzzi, F., De Silva, L.: Planning in bdi agents: a survey of the integration of planning algorithms and agent reasoning. *The Knowledge Engineering Review* **30**(1), 1–44 (2015). <https://doi.org/10.1017/S0269888913000337>
77. Mialon, G., Dessì, R., Lomeli, M., Nalmpantis, C., Pasunuru, R., Raileanu, R., Rozière, B., Schick, T., Dwivedi-Yu, J., Celikyilmaz, A., et al.: Augmented language models: a survey. arXiv preprint arXiv:2302.07842 (2023)
78. Miller, T.: Explanation in artificial intelligence: Insights from the social sciences. *Artif. Intell.* **267**, 1–38 (2019). <https://doi.org/10.1016/j.artint.2018.07.007>
79. Muller, G., Maneerat, S., Pajot, H., Petridis, C., Niamke, R.: Using (traditional) multi-agent principles to suggest improvements for agentic ai platforms (2025), Agent Toolkits 2025. Available at <https://agent-toolkits.github.io/Agent-Toolkits-2025/papers/muller-et-al.pdf>
80. Müller, J.P., Fischer, K.: Application impact of multi-agent systems and technologies: A survey. In: Shehory, O., Sturm, A. (eds.) *Agent-Oriented Software Engineering: Reflections on Architectures, Methodologies, Languages, and Frameworks*, pp. 27–53. Springer Berlin Heidelberg, Berlin, Heidelberg (2014). https://doi.org/10.1007/978-3-642-54432-3_3, https://doi.org/10.1007/978-3-642-54432-3_3
81. Nallur, V.: Social simulation and multi-agent systems: A false dichotomy in the age of generative agents (2025), Agent Toolkits 2025. Available at <https://agent-toolkits.github.io/Agent-Toolkits-2025/papers/nallur.pdf>
82. Nallur, V., Cardozo, N., Clarke, S.: Clonal plasticity: a method for decentralized adaptation in multi-agent systems. In: Proceedings of the 11th international symposium on software engineering for adaptive and self-managing systems. pp. 122–128. ACM (2016). <https://doi.org/10.1145/2897053.2897067>, <http://www.tara.tcd.ie/bitstream/handle/2262/78715/p122-nallur.pdf?sequence=1&isAllowed=y>
83. Nallur, V., O’Toole, E., Cardozo, N., Clarke, S.: Algorithm diversity: A mechanism for distributive justice in a socio-technical MAS. In: Proceedings of the 2016 international conference on autonomous agents & multiagent systems. pp. 420–428. International Foundation for Autonomous Agents and Multiagent Systems (2016), <https://www.ifaamas.org/Proceedings/aamas2016/pdfs/p420.pdf>
84. Naudet, Y., Panetto, H., Yilma, B.A.: Towards cognitive interoperability in cyber-physical enterprises. *IFAC-PapersOnLine*, 22nd IFAC World Congress **56**(2), 695–706 (2023). <https://doi.org/https://doi.org/10.1016/j.ifacol.2023.10.1648>

85. Newell, A.: The knowledge level. *Artif. Intell.* **18**(1), 87–127 (jan 1982). [https://doi.org/10.1016/0004-3702\(82\)90012-1](https://doi.org/10.1016/0004-3702(82)90012-1), [https://doi.org/10.1016/0004-3702\(82\)90012-1](https://doi.org/10.1016/0004-3702(82)90012-1)
86. Ngo, R., Chan, L., Mindermann, S.: The alignment problem from a deep learning perspective (2025), <https://arxiv.org/abs/2209.00626>
87. Olivier Boissier, Rafael H. Bordini, J.H., Ricci, A.: *Multi-Agent Oriented Programming, Programming Multi-Agent Systems Using JaCaMo*. MIT Press (2020)
88. O’Toole, E., Nallur, V., Clarke, S.: Decentralised detection of emergence in complex adaptive systems. *ACM Transactions on Autonomous and Adaptive Systems* **12**(1), 1–31 (2017). <https://doi.org/10.1145/3019597>
89. Patterson, D., Gonzalez, J., Le, Q., Liang, C., Munguia, L.M., Rothchild, D., So, D., Texier, M., Dean, J.: Carbon emissions and large neural network training (2021), <https://arxiv.org/abs/2104.10350>
90. Poslad, S.: Specifying protocols for multi-agent systems interaction. *ACM Trans. Auton. Adapt. Syst.* **2**(4), 15–es (Nov 2007). <https://doi.org/10.1145/1293731.1293735>
91. Qin, Y., Hu, S., Lin, Y., Chen, W., Ding, N., Cui, G., Zeng, Z., Zhou, X., Huang, Y., Xiao, C., et al.: Tool learning with foundation models. *ACM Computing Surveys* **57**(4), 1–40 (2024)
92. Qin, Y., Liang, S., Ye, Y., Zhu, K., Yan, L., Lu, Y., Lin, Y., Cong, X., Tang, X., Qian, B., et al.: ToolLLM: Facilitating Large Language Models to Master 16000+ Real-world APIs. *arXiv preprint arXiv:2307.16789* (2023)
93. Qu, C., Dai, S., Wei, X., Cai, H., Wang, S., Yin, D., Xu, J., Wen, J.R.: Tool learning with large language models: A survey. *Frontiers of Computer Science* **19**(8), 198343 (2025)
94. Rafailov, R., Sharma, A., Mitchell, E., Manning, C.D., Ermon, S., Finn, C.: Direct preference optimization: Your language model is secretly a reward model. In: Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S. (eds.) *Advances in Neural Information Processing Systems*. vol. 36, pp. 53728–53741. Curran Associates, Inc. (2023)
95. Rao, A.S.: Agentspeak(1): Bdi agents speak out in a logical computable language. In: Van de Velde, W., Perram, J.W. (eds.) *Agents Breaking Away*. pp. 42–55. Springer Berlin Heidelberg, Berlin, Heidelberg (1996)
96. Rao, A.S., Georgeff, M.P.: BDI agents: From theory to practice. In: *ICMAS*. pp. 312–319. The MIT Press (1995)
97. Ricci, A., Ciortea, A.: Empowering the concept of tool in Agentic AI: Co-evolving AI agents and environments with A&A and Hypermedia MAS (2025), *Agent Toolkits 2025*. Available at <https://agent-toolkits.github.io/Agent-Toolkits-2025/papers/ricci-ciortea.pdf>
98. Ricci, A., Piunti, M., Viroli, M.: Environment programming in multi-agent systems: an artifact-based perspective. *Autonomous Agents and Multi-Agent Systems* **23**(2), 158–192 (Sep 2011). <https://doi.org/10.1007/s10458-010-9140-7>, <https://doi.org/10.1007/s10458-010-9140-7>
99. Ricci, A., Viroli, M., Omicini, A.: Programming MAS with Artifacts. In: Bordini, R.H., Dastani, M.M., Dix, J., El Fallah Seghrouchni, A. (eds.) *Programming Multi-Agent Systems*. pp. 206–221. Springer Berlin Heidelberg, Berlin, Heidelberg (2006)
100. Rodriguez, S., Thangarajah, J.: Explainable agents (XAg) by design (blue sky ideas track). In: Alechina, N., Dignum, V., Dastani, M., Sichman, J. (eds.) *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. ACM (2024)

101. Rodriguez, S., Thangarajah, J., Davey, A.: Design patterns for explainable agents (XAg). In: Alechina, N., Dignum, V., Dastani, M., Sichman, J. (eds.) Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS). ACM (2024)
102. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach. Pearson, 4 edn. (2022)
103. Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., Scialom, T.: Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems* **36**, 68539–68551 (2023)
104. Schoen, B., Nitishinskaya, E., Balesni, M., Højmark, A., Hofstätter, F., Scheurer, J., Meinke, A., Wolfe, J., van der Weij, T., Lloyd, A., Goldowsky-Dill, N., Fan, A., Matveikin, A., Shah, R., Williams, M., Glaese, A., Barak, B., Zaremba, W., Hobbhahn, M.: Stress testing deliberative alignment for anti-scheming training (2025), <https://arxiv.org/abs/2509.15541>
105. Shen, Q., Polyvyanyy, A., Lipovetzky, N., Kampik, T.: Agent system event data: Concepts, dimensions, applications. In: Maass, W., Han, H., Yasar, H., Multari, N.J. (eds.) *Conceptual Modeling - 43rd International Conference, ER 2024, Pittsburgh, PA, USA, October 28-31, 2024, Proceedings. Lecture Notes in Computer Science*, vol. 15238, pp. 56–72. Springer (2024). https://doi.org/10.1007/978-3-031-75872-0_4
106. Shoham, Y.: Agent-oriented programming. *Artificial Intelligence* **60**(1), 51–92 (1993). [https://doi.org/https://doi.org/10.1016/0004-3702\(93\)90034-9](https://doi.org/https://doi.org/10.1016/0004-3702(93)90034-9), <https://www.sciencedirect.com/science/article/pii/0004370293900349>
107. Shojaei, P., Mirzadeh, I., Alizadeh, K., Horton, M., Bengio, S., Farajtabar, M.: The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *arXiv preprint arXiv:2506.06941* (2025)
108. da Silva, J.L.T., Demazeau, Y.: Vowels co-ordination model. In: *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 3*. p. 1129–1136. AAMAS '02, Association for Computing Machinery, New York, NY, USA (2002). <https://doi.org/10.1145/545056.545083>
109. Singh, M.P.: Agent communication languages: Rethinking the principles. *IEEE Computer* **31**(12), 40–47 (Dec 1998)
110. Singh, M.P.: Information-driven interaction-oriented programming: BSPL, the Blindly Simple Protocol Language. In: *Proceedings of the 10th International Conference on Autonomous Agents and MultiAgent Systems*. pp. 491–498. IFAA-MAS (2011)
111. Singh, M.P., Christie, S.H., Chopra, A.K.: Langshaw: Declarative interaction protocols based on sayso and conflict. In: *Proceedings of the 33rd International Joint Conference on Artificial Intelligence (IJCAI)*. pp. 202–210. IJCAI, Jeju, Korea (Aug 2024). <https://doi.org/10.24963/ijcai.2024/23>
112. Song, H., Elgammal, A., Nallur, V., Chauvel, F., Fleurey, F., Clarke, S.: On architectural diversity of dynamic adaptive systems. In: *2015 IEEE/ACM 37th IEEE international conference on software engineering*. vol. 2, pp. 595–598 (2015). <https://doi.org/10.1109/ICSE.2015.201>
113. Song, Y., Xiong, W., Zhu, D., Wu, W., Qian, H., Song, M., Huang, H., Li, C., Wang, K., Yao, R., et al.: Restgpt: Connecting large language models with real-world restful APIs. *arXiv preprint arXiv:2306.06624* (2023)
114. Sorici, A., Olaru, A.: Experience-based agent communities in support of smart environment hypermedia MAS (2025), *Agent Toolkits 2025*. Available at <https://agent-toolkits.github.io/Agent-Toolkits-2025/papers/sorici-olaru.pdf>

115. Sturm, A., Shehory, O.: Agent-oriented software engineering: Revisiting the state of the art. In: Shehory, O., Sturm, A. (eds.) *Agent-Oriented Software Engineering: Reflections on Architectures, Methodologies, Languages, and Frameworks*, pp. 13–26. Springer Berlin Heidelberg, Berlin, Heidelberg (2014). https://doi.org/10.1007/978-3-642-54432-3_2
116. Sumers, T., Yao, S., Narasimhan, K., Griffiths, T.: Cognitive architectures for language agents. *Transactions on Machine Learning Research* (2024), <https://openreview.net/forum?id=1i6ZCvflQJ>, survey Certification
117. Surapaneni, R., Jha, M., Vakoc, M., Segal, T.: Announcing the Agent2Agent Protocol (A2A). <https://developers.googleblog.com/en/a2a-a-new-era-of-agent-interoperability/> (2025), accessed: 28.04.2025
118. Tennant, E., Hailes, S., Musolesi, M.: Moral alignment for llm agents (2025), <https://arxiv.org/abs/2410.01639>
119. Ullrich, J., Koch, M., Vogelsang, A.: From requirements to code: Understanding developer practices in LLM-assisted software engineering (2025), <https://arxiv.org/abs/2507.07548>
120. Vachtsevanou, D., Ciortea, A., Mayer, S., Lemée, J.: Signifiers as a first-class abstraction in hypermedia multi-agent systems. In: *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*. pp. 1200–1208 (2023)
121. Vachtsevanou, D., Lemée, J., Tamma, V., Payne, T.R., Ciortea, A., Mayer, S.: A vision for llm-based interaction assistance for autonomous agents on the semantic web. In: Ciortea, A., Collier, R., Gandon, F., Harth, A., Zimmermann, A. (eds.) *The Second International Workshop on Hypermedia Multi-Agent Systems (HyperAgents 2025)*. pp. 56–65. No. 994 in *CEUR Workshop Proceedings, Aachen* (2025), <https://ceur-ws.org/Vol-4084/short7.pdf>, To appear.
122. Valmeekam, K., Stechly, K., Kambhampati, S.: LLMs Still Can’t Plan; Can LRMs? A Preliminary Evaluation of OpenAI’s o1 on PlanBench (Sep 2024). <https://doi.org/10.48550/arXiv.2409.13373>, arXiv:2409.13373 [cs]
123. W3C: Web of Things (WoT) thing description 1.1, W3C candidate recommendation. <https://www.w3.org/TR/wot-thing-description/> (2025), accessed: 2025-10-20
124. Wang, Y., Pan, Y., Su, Z., Deng, Y., Zhao, Q., Du, L., Luan, T.H., Kang, J., Niyato, D.: Large model based agents: State-of-the-art, cooperation paradigms, security and privacy, and future trends. *IEEE Communications Surveys and Tutorials* (2025). <https://doi.org/10.1109/COMST.2025.3576176>
125. Wang, Z., Cheng, Z., Zhu, H., Fried, D., Neubig, G.: What are tools anyway? a survey from the language model perspective. arXiv preprint arXiv:2403.15452 (2024)
126. Weidinger, L., Mellor, J., Rauh, M., Griffin, C., Uesato, J., Huang, P.S., Cheng, M., Glaese, M., Balle, B., Kasirzadeh, A., Kenton, Z., Brown, S., Hawkins, W., Stepleton, T., Biles, C., Birhane, A., Haas, J., Rimell, L., Hendricks, L.A., Isaac, W., Legassick, S., Irving, G., Gabriel, I.: Ethical and social risks of harm from language models (2021), <https://arxiv.org/abs/2112.04359>
127. Weyns, D., Omicini, A., Odell, J.: Environment as a first class abstraction in multiagent systems. *Autonomous Agents and Multi-Agent Systems* **14**(1), 5–30 (Feb 2007). <https://doi.org/10.1007/s10458-006-0012-0>
128. Weyns, D., Steegmans, E., Holvoet, T.: Towards active perception in situated multi-agent systems. *Applied Artificial Intelligence* **18**(9-10), 867–883 (2004). <https://doi.org/10.1080/08839510490509063>, <https://doi.org/10.1080/08839510490509063>

129. Winikoff, M.: Challenges and Directions for Engineering Multi-agent Systems. CoRR **abs/1209.1428** (2012), <http://arxiv.org/abs/1209.1428>, arXiv: 1209.1428
130. Winikoff, M.: 30 Years of Engineering Multi-Agent Systems: What and Why? In: Dastani, M., Sichman, J.S., Alechina, N., Dignum, V. (eds.) Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2024, Auckland, New Zealand, May 6-10, 2024. p. 3. International Foundation for Autonomous Agents and Multiagent Systems / ACM (2024). <https://doi.org/10.5555/3635637.3662844>
131. Winikoff, M.: Towards using LLMs to (co-)create agent(ic) systems (2025), Agent Toolkits 2025. Available at <https://winikoff.github.io/pdf/llm.pdf>
132. Winikoff, M., Sidorenko, G., Dignum, V., Dignum, F.: Why bad coffee? Explaining BDI agent behaviour with valuing. Artif. Intell. **300**, 103554 (2021). <https://doi.org/10.1016/J.ARTINT.2021.103554>