Dagstuhl Seminar 23081 on "Agents on the Web"

(February 19-24, 2023, Schloss Dagstuhl, Warden, Germany)

Organizers: Olivier Boissier, Andrei Ciortea, Andreas Harth, Alessandro Ricci

***Abstract*—The World Wide Web has emerged as the middleware of choice for most distributed systems. Recent standardization efforts for the Web of Things and Linked Data are now turning hypermedia into a homogeneous information fabric that interconnects everything — devices, information resources, abstract concepts, etc. The latest standards allow clients not only to *browse* and *query*, but also to *observe* and *act on* this hypermedia fabric. Researchers and practitioners are already looking for means to build more sophisticated clients able to meet their design objectives through *flexible autonomous use* of this hypermedia fabric. Such autonomous agents have been studied to large extent in research on distributed artificial intelligence and, in particular, multi-agent systems. These recent developments thus motivate the need for a broader perspective that can only be achieved through a concerted effort of the research communities on *the Web Architecture and the Web of Things*, *Semantic Web and Linked Data*, and *Autonomous Agents and Multi-Agent Systems*. The primary objective of this seminar is to support the transfer of knowledge and results across these communities in order to pave the way for a new generation of autonomous systems on the Web. We believe this seminar can break new ground in all three areas of research.**

I. Research Areas

The vision of autonomous agents on the Web is almost as old as the Web itself: in his keynote at WWW'94[1], Sir Tim Berners-Lee was noting that documents on the Web describe real objects and relationships among them, and if the semantics of these objects are represented explicitly then machines can *browse through* and *manipulate* reality.[2] These ideas were published under the Semantic Web vision in 2001 [2]. Yet in 2007, after having spent the better half of a decade advancing this vision, James Hendler was looking back to conclude that most ideas in the original article were already seeing widespread deployment on the Web except for agent-based systems — and raised the question: "where are all the intelligent agents?" [27].

This question is yet to be addressed. On today's Web we are often assisted by invisible software agents, such as crawlers used by search engines to navigate and index Web pages [4], agents that curate online content produced by people (e.g., Wikipedia's content agents [41]), and recommender systems used all over the Web to generate more links and navigation paths (e.g., suggestions of books on an Amazon page). In our everyday lives we are assisted by more visible agents, such as Amazon's Alexa, Google Duplex, or Apple's Siri. Some of these agents may already use various AI methods (learning, reasoning, etc.), but they are specialized for narrow tasks and constrained to silos dictated by company ecosystems. We have yet to see more autonomous, cooperative, and long-lived agents on the Web [28] — the intelligent agents in James Hendler's question. We believe this decade-old question is now more relevant than ever before and can be better addressed in the context of recent developments in three areas of research: (i) *Web Architecture and the Web of Things*, (ii) *Semantic Web and Linked Data*, and (iii) *Autonomous Agents and Multi-Agent Systems*.

The primary objective of this seminar is to revisit and align the relevant research threads in the above-mentioned communities. We discuss in more detail each of these areas in what follows and highlight why an integration across these areas would be beneficial. We believe that, together, these communities could pave the way for a new generation of autonomous systems on the Web.

A. Web Architecture and the Web of Things

Recent standardization efforts for the Web of Things (WoT) [24][3] allow constrained Web servers to target devices with as little as 10 KiB of RAM and 100 KiB of ROM[4], which means sensors and actuators can now be abstracted behind embedded Web services. The World Wide Web is turning into the middleware of choice for most systems envisioned on the Internet and its huge success comes from its carefully designed architectural properties.

A central feature of REST, the architectural style of the Web, is that it uses hypermedia to drive the interaction between components, principle known as *hypermedia as the engine of application state (HATEOAS)* — see [17] for details. To illustrate this principle, an HTML page typically provides the user with a number of *affordances*[5], such as to navigate to a different page by clicking a hyperlink or to submit an order by filling out and submitting an HTML form. Performing any such action transitions the application to a new state, which in turn provides the user with a new set of affordances. In each state, the user's browser retrieves not only an HTML representation of the current state from a server, but also a selection of next possible states and the information required to construct the HTTP requests that can be issued to transition to those states. Retrieving all this information through hypermedia allows the Web application to evolve without impacting the browser, and allows the browser to transition seamlessly

[1]The First International Conference on the World-Wide Web, CERN, 25 - 27 May, 1994.

[2]Sir Tim Berners-Lee, *The Future of the Web*, WWW'94: https://videos.cern.ch/record/2671957, accessed: 20.12.2020.

[3]The WoT is currently being standardized through combined efforts of the W3C (https://www.w3.org/WoT/) and the IETF (https://datatracker.ietf.org/wg/core/about/).

[4]The *Constrained Application Protocol (CoAP)* [46], a Web transfer protocol for constrained nodes and networks, was designed to target *Class 1 devices* (see IETF terminology for constrained-node networks [3]).

[5]*Affordances* are used in this context to denote *interaction cues*. The term is inspired from *affordance theory* [42].

across servers. HATEOAS thus reduces coupling between Web components (e.g., browsers, origin servers, intermediaries) and allows them to be deployed and to evolve independently from one another — a central feature that allowed the Web to evolve into a world-wide, open, and long-lived system.

Most existing Web services do not use hypermedia and its benefits in terms of loose coupling. Over the past two decades, however, Web service design has shifted from a predominantly RPC-style paradigm towards a REST-style paradigm. The RPC-style paradigm would typically use the Web as a transport layer — for instance, to transport serializations of procedure calls via HTTP. Prominent examples of RPC-style Web service design include the WS-* standards (SOAP, WSDL, UDDI, etc.), which use HTTP to transport SOAP messages [38], and the more recent gRPC[6]. Today, it is well recognized that systems using the Web merely as a transport layer remain outside of the Web[7] [43] and are misaligned with REST (see Section 6.5.3 in [16]). Depending on a system's design objectives, this is not an inherent problem, but it also means such systems cannot fully benefit from the properties of the Web in terms of scalability, openness, and evolvability. Over the past decade, REST-like designs have become the *de facto* industry standard for Web services, but in most cases such designs do not use hypermedia and HATEOAS: instead they typically rely on static Web APIs that implement create, read, update, and delete (CRUD) operations over HTTP.

In recent years, hypermedia and HATEOAS have been gaining momentum in Web service design (e.g., [30], [34], [52]). The explosive growth in the number of Web APIs[8] leads to increasingly complex systems for which the use of static Web APIs becomes impractical: developers have to manually integrate an increasing number of heterogeneous Web APIs across service providers — and then to maintain those integrations as the Web APIs evolve over time. This is particularly a problem in WoT systems with constrained devices that are often duty-cycled, or with mobile devices that physically move between spatial domains together with the (localized) services they provide: the dynamic and open nature of these systems requires components to be deployed and to evolve independently from one another. To address these challenges, researchers and practitioners have turned to HATEOAS as a means to design dynamic Web APIs that expose hypermedia affordances to clients (e.g., see the W3C WoT Thing Description [30], Hydra [34], RESTdesc [52]) — and are now looking for means to design more sophisticated autonomous clients able to discover and use hypermedia affordances at run time (e.g., [1], [33], [37], [9], [11]).

B. Semantic Web and Linked Data

Tim Berners-Lee published the first blueprint architecture for the Semantic Web in 1998.[9] Since the early 2000s, the Semantic Web has grown into a well-established area of research that covers a broad range of topics (see [20] for a survey of the main research threads in this area). Two of the oldest and still very active research topics are (i) *knowledge representation and reasoning* based on logical formalisms, and (ii) *ontologies and semantic vocabularies* on the Web. Much of the work focused around these interrelated topics was influenced by early research on Artificial Intelligence (AI), knowledge acquisition, and knowledge-based systems (see also Section I-C).

An important application of the above-mentioned research topics, Semantic Web services can also be traced back to the early 2000s. Over the years, a variety of solutions have been proposed to describe Web services — for instance, using languages for the syntactic description of interfaces (e.g., WSDL [7], WADL [25], OpenAPI[10], RAML[11]), languages for the semantic annotation of interfaces (e.g., SAWSDL [32], SA-REST [35]), and ontologies for Web services (e.g., OWL-S [36], WSMO [45]). Clients can then interpret and reason on semantic descriptions of services in order to discover, invoke, and compose services.[12] The more recent work on hypermedia-driven Web APIs discussed in Section I-A draws from this line of research.

The above-mentioned research topics focus mainly on logical formalisms and reasoning. An important step in the evolution of the Semantic Web came in 2006, when Tim Berners-Lee published the four rules of Linked Data to stress that "the Semantic Web isn't just about putting data on the web. It is about making links, so that a person or machine can explore the web of data".[13] Linked Data brought more focus on the use of URIs as a mechanism to discover and retrieve resources on the Semantic Web and can be seen as the first deployment wave of the Semantic Web vision [20]: it led to the publication of a wealth of linked open datasets that clients can consume in a uniform manner.[14]

Various approaches are now available to expose Linked Data in RDF. The most common approach is to use SPARQL endpoints [26]. SPARQL endpoints provide clients with low-cost and fine-grained access to RDF data, but they are also known for low availability and poor server-side performance [5]: the workload for executing SPARQL queries and updates is offloaded on the server-side. As an alternative, Triple Pattern Fragments (TPF) interfaces [53] use HATEOAS to distribute the workload between clients and servers: clients use hypermedia affordances to discover and retrieve Linked Data Fragments of various granularity levels, and then process the data on the client-side.

SPARQL endpoints and TPF interfaces are designed primarily for querying Linked Data. Recent W3C recommendations such as the Linked Data Platform [49] and Linked Data Notifications [6] allow automated clients to *browse*, *observe*, and *act on* Linked Data on the Web in a standard and uniform

[6]https://grpc.io/, accessed: 20.12.2020.

[7]E.g., the SOAP specification also defines a binding for sending messages via SMTP [38] — and thus completely outside of the Web.

[8]https://bit.ly/2lEWvmY, accessed: 20.12.2020.

[9]https://www.w3.org/DesignIssues/Semantic.html

[10]https://www.openapis.org/

[11]https://raml.org/

[12]Web service composition was strongly influenced by research on *automated planning*. A survey of this large body of work is available in [47].

[13]https://www.w3.org/DesignIssues/LinkedData.html

[14]As of May 2020, the *Linked Open Data cloud* contains 1269 datasets and 16201 links: https://lod-cloud.net/.

manner. Linked Data now creates the underpinning of the Semantic Web architecture. Together with open specifications such as SOLID[15], these recent developments promote the decentralization of the Semantic Web [51] and peer-to-peer interactions on the Web. Some researchers are now returning to the original Semantic Web vision and James Hendler's question, looking for new paradigms and programming languages for more sophisticated Linked Data clients — for instance, using reactive rule-based programming (e.g., [50], [31]) or dedicated languages for defining and embedding (dereferenceable) functions in RDF graphs (e.g., using LDScript [12]).

C. Autonomous Agents and Multi-Agent Systems

Much of the research on autonomous agents, automated reasoning and planning can be traced back to the mid-80s [21] or even the early 70s [18], to the seminal work conducted at the Stanford Research Institute. In AI research, an *agent* is commonly defined as "a computer system, situated in some environment, that is capable of flexible autonomous action in order to meet its design objectives" [29]. Autonomy is central to this definition and refers to the agent's ability to operate on its own, without the need of direct intervention from people or other agents. The agent is typically situated in an external environment that it can perceive via sensors and influence via actuators.[16] In distributed AI, a Multi-Agent System (MAS) is then a system conceptualized in terms of agents situated in a shared environment that interact with one another to meet their design objectives [54].

Many of the underlying research questions the previous two communities are now confronted with — such as: *how to engineer autonomous software agents*, *how to balance reactive and goal-directed behavior in software agents*, or *how to govern autonomous behavior* — have been investigated in the scientific literature on autonomous agents and MAS [54]. On the other hand, the MAS community is posed with an important challenge: agent technologies have not yet achieved their long-standing promise to enter mainstream software engineering [40]. The previous communities can thus benefit from results in MAS research and at the same time unlock new practical use cases for MAS, in particular in the context of WoT and Linked Data systems (see also [10]). Going further, another long-standing challenge in MAS research is to create world-wide and long-lived MAS: in 2001, the *Agentcities* initiative was aiming to create a world-wide open network of heterogeneous agents to which any organization or individual researcher could connect their agents [57]. We have yet to witness the deployment of such systems. The Web, however, is the most scalable and versatile software system deployed on the Internet to date — and research on the Web architecture could now provide new insight into engineering large-scale, open, and long-lived MAS (e.g., see [8]).

[15]https://solidproject.org/, accessed: 20.12.2020.

[16]A distinctive feature of an autonomous agent is its flexibility in the pursuit of some design objectives [29]: the agent is *reactive* by responding to changes in the environment in a timely fashion, *proactive* by exhibiting goal-directed behavior and taking the initiative when appropriate, and *social* by interacting with humans or other agents in order to achieve complex tasks that would otherwise overcome its own capabilities.

The Web has already raised interest in MAS research, in particular with the advent of service-oriented computing in the early 2000s. Most notably, Munindar Singh and Michael Huhns have argued for the necessity of a broader perspective to study service-oriented computing — one that would include MAS [48]. Most of the early research was influenced by *service-oriented architectures* based on the main standards for Web services available at the time (the WS-* standards: SOAP, WSDL, UDDI etc.), which only use the Web as a transport layer (see Section I-A). The Foundation for Intelligent Physical Agents (FIPA) also proposed a specification for using HTTP as a transport protocol for messages exchanged among agents [19], which was implemented by several FIPA-compliant platforms (e.g., [23], [13], [15]). As discussed in Section I-A, systems using the Web only as a transport layer do not fully benefit from its architectural properties and existing infrastructure. More recent approaches have turned to *resource-oriented architectures* based on REST-like Web services that implement CRUD operations over HTTP (e.g., [39], [22]), but they generally do not use hypermedia and HATEOAS. Two exceptions are [8] and [14]: the former uses HATEOAS to design Web-based MAS and the latter applies the Linked Data principles (which partly reflect HATEOAS) to bring autonomous agents on the Web. Both approaches consider the agents' environment as *a first-class abstraction* in MAS (see also [56], [44]) and a means to deploy MAS on the Web.

Until more recently, the agents' environment was not considered a first-class abstraction in MAS. In the conventional view, MAS are only composed of agents, which leaves little room for the Web as anything more than a transport layer for messages exchanged among agents. Over the past decade, however, it has become apparent that MAS can consist of more than just agents and thus should be designed on multiple dimensions. Communities such as *Environment for Multiagent Systems (E4MAS)*[17] [55] and *Coordination, Organizations, Institutions, and Norms in Agent Systems (COIN)*[18] have advanced the *environment* and the *organization* (respectively) as first-class abstractions in MAS. These conceptual dimensions open new perspectives on the conceptual integration of MAS and the Web [10].

If we conceive of the *environment* as a first-class abstraction in MAS then the Web is no longer a hidden transport layer for agent messages, but a visible application layer and a place for stigmergic interactions — a world-wide environment that can be designed and programmed for the agents. Web resources regain their status as first-class abstractions situated in the agents' environment, where agents can share, observe, reason about and act upon them. At the same time, MAS no longer remain outside of the Web, but instead are weaved into the Web. The *organization* dimension is equally relevant for achieving a proper conceptual integration of MAS and the Web. For instance, the various normative and organizational models proposed by the COIN community over the past decade could potentially be used to represent norms, orga-

[17]https://distrinet.cs.kuleuven.be/events/e4mas/, accessed: 20.12.2020.

[18]https://www2.pcs.usp.br/~coin/coin_springer.html, accessed: 20.12.2020.

nizational structures, and social relations in the environment (externally to the agents). Governance and social enforcement mechanisms could then allow to control and regulate the autonomous behavior of agents on the Web. The alternative of hard-coding rules into agents (e.g., terms of service, data licensing policies) would be impractical in an environment as open and complex as the Web.

The recent developments towards hypermedia-driven Web APIs, the WoT, and Linked Data are turning *hypermedia* into a homogeneous information fabric that interconnects everything: devices, information resources, abstract concepts, etc. The latest W3C recommendations as well as various open standards allow automated clients to reliably *browse*, *query*, *observe*, and *act on* this hypermedia fabric. Researchers and practitioners are already looking for means to build more sophisticated clients able to meet their design objectives through *flexible autonomous use* of this hypermedia fabric. Such autonomous agents have been investigated in MAS research, and the roots in logical formalisms shared by both MAS and Semantic Web research further simplify the alignment of these areas. All these recent developments bring new insight into creating a conceptual bridge between MAS and the Web [10]. This state of affairs motivates the need for a broader perspective, which can only be achieved through a concerted effort of the targeted communities.

II. Objectives and Expected Outcome

The overall objective of this seminar is to support the transfer of knowledge and results across the targeted communities (see Section I). Concretely, the organization of the seminar will pursue the following sub-objectives:

- O1 to identify and align the various research threads in the targeted communities that are relevant for advancing the research on autonomous agents on the Web;
- O2 to work towards a shared conceptualization and theoretical framework for autonomous agents on the Web;
- O3 to identify representative use cases in different domains that would help demonstrate the potential impact of this joint research effort on society and economy;
- O4 to evaluate the state of technologies available for prototyping and deploying autonomous agents on the Web (and to identify any potential gaps).

The seminar aims to deliver results focused on achieving objectives O1-O4:

- R1 a consolidated review (across the three targeted areas) of the relevant research threads identified in O1;
- R2 a research agenda for autonomous agents on the Web with coordinated / joint action items across the three targeted communities (cf. O2);
- R3 a set of representative use cases for autonomous agents on the Web (cf. O3);
- R4 a consolidated review (across the three targeted areas) of relevant technologies that are already available (cf. O4);
- R5 a technological roadmap for autonomous agents on the Web with coordinated / joint action items across the three targeted communities (cf. O4).

The above results are to be presented in a technical report that will be drafted during the seminar with the participation of all invitees. The co-organizers will then consolidate the technical report after the seminar.

III. Research Questions

In what follows, we propose four main research questions to be pursued during the seminar. This list is not exhaustive, it is only meant to kick off the discussions. We will also cater for research questions that emerge during our discussions at the seminar.

RQ1: How to design software agents able to achieve their tasks through flexible autonomous use of hypermedia?

Such *autonomous hypermedia agents* would be able to *navigate* the hypermedia to *discover*, *reason*, *observe*, and *act on* resources required to achieve their tasks. Among others, this research question includes sub-questions such as:

- How to *describe* and *reliably identify* autonomous agents on the Web?
- What *type of knowledge* does an autonomous hypermedia agent require?
- What *architectures* are suitable to design autonomous hypermedia agents?
- What *programming paradigms* and *languages* are suitable to program autonomous hypermedia agents?

RQ2: How to design hypermedia-based environments that support autonomous behavior?

In AI research, the *autonomy* of an agent is defined in relationship to the agent's environment, which the agent can *perceive* and *act upon* (see Section I-C). Among others, this research question includes sub-questions such as:

- How to enable *effective navigation and search* in *Web-scale* hypermedia-based environments?
- How to define and represent *perception* in hypermedia-based environments on the Web?
- How to define and represent *action* in hypermedia-based environments on the Web?
- How to *model* and *program* hypermedia-based environments for autonomous agents?

RQ3: How to design, represent, and reason about interactions among autonomous agents, people, and any other resources on the Web?

While RQ1 and RQ2 focus on the agent-level perspective, RQ3 zooms out on the system-level perspective — accounting also for people on the Web. Among others, this research question includes sub-questions such as:

- How to define *interaction protocols* for both autonomous agents and people on the Web?
- How to *represent* and *reason about* interactions and interaction protocols on the Web?
- How to make autonomous interactions on the Web *transparent* and *explainable* to people?
- How to design and visualize Web-based *mixed-reality interactions* among autonomous agents and people?

RQ4: How to design and govern communities of autonomous agents and people on the Web?

While RQ2 and RQ3 focus on enabling *autonomous interactions on the Web*[19], RQ4 focuses on the challenges such interactions bring. Among others, this research question includes sub-questions such as:

- How to *represent* and *reason about* policies and norms on the Web (terms of service, data licensing policies, user preferences, etc.)?
- How to *capture* and *reason about* social constructs on a Web populated with autonomous agents?
- How to *coordinate*, *monitor*, and *regulate* interactions in *Web-scale communities* of autonomous agents and people?
- How to *preserve privacy* on a Web populated with autonomous agents?

IV. Composition of the Organizing Team

The organizing team brings together recognized researchers with complementary expertise across the three targeted areas. Prof. Olivier Boissier and Prof. Alessandro Ricci have been involved in the two initiatives establishing the *organization* and the *environment* (respectively) as first-class abstractions in MAS (see Section I-C). Prof. Andreas Harth has contributed to searching and querying Linked Data at Web scale, and more recently he has been advancing the research on modeling and executing dynamical behavior on the Web. Prof. Dr. Andrei Ciortea is contributing to defining a new class of Web-based MAS that can inherit the architectural properties of the Web and preserve the properties of MAS. His work has been published in both the AAMAS and WoT/IoT communities.

All four co-organizers have rich experience in organizing academic events and projects (see biographies below). Dr. Andrei Ciortea and Prof. Olivier Boissier are also two of the initiators of the *First Workshop on Hypermedia Multi-Agent Systems (HyperAgents 2019)*[20], which was co-located with The Web Conference 2019 – formerly known as WWW, the flagship conference of the World Wide Web community. The HyperAgents workshops aim to establish a common forum for the Web and MAS communities, and to create social, conceptual, and technological bridges across these fields.

A. Olivier Boissier

Prof. Dr. Olivier Boissier is a full professor of computer science at Mines Saint-Etienne (France). He is member of the Institut Henri Fayol at Mines Saint-Etienne and of Hubert Curien Laboratory UMR CNRS 5516 where he leads the Connected Intelligence Research Team. He is active in the research and development of multi-agent systems. His main research contributions concern: coordination and control of multi-agent systems, multi-agent oriented programming. MSc and PhD theses that he has advised have contributed to the development of MOISE organizational models, agents' architectures, autonomy and control, JaCaMo platform. They all contribute to the definition of multi-agent models, tools and methodologies for the multi-agent oriented engineering of software applications in the smart industry and smart city domains. Olivier Boissier is serving on the program committees of many conferences (IAT/WI, AAMAS) and workshops. He was the program chair of the IAT/WI conference in 2011. O. Boissier has presented tutorials on Organization Oriented Programming at AAMAS 2008 and 2004, EASSS 2007, 2005 on Multi-Agent Oriented Programming at AAMAS 2015, EASSS 2012, 2011, 2010.

B. Andrei Ciortea

Prof. Dr. Andrei Ciortea is an assistant professor in Computer Science with a focus on Web-based Systems at the School of Computer Science, University of St. Gallen (HSG), Switzerland, and an external collaborator of the Wimmics team at Inria, Université Côte d'Azur, CNRS, I3S, France. His main research interests include Web-based multi-agent systems (MAS), hypermedia systems, the Web of Things (WoT), and socio-technical networks. Dr. Ciortea received the best paper award at the 6th International Conference on the Internet of Things (IoT 2016), and since 2017 he has published regularly in the AAMAS and Engineering MAS communities. Prior to joining HSG, he had a 6-month research visit with Siemens' Web of Systems research group in Berkeley, California, where he was the principal investigator and lead architect of a system for intelligent manufacturing that integrated MAS with WoT systems. Dr. Ciortea joined the W3C WoT Working Group in 2016 and is a founding member of the *Romanian Association for Artificial Intelligence (ARIA)*. In ARIA, he served as a member of the board of directors from 2011 to 2016, during which time he (co-)initiated 2 international and 7 national projects.

C. Andreas Harth

Prof. Dr. Andreas Harth is a professor of information systems at Friedrich-Alexander-University Erlangen-Nuremberg and Department Leader at Fraunhofer IIS-SCS. Prior to moving to Nuremberg, Dr. Harth worked as a post-doctoral researcher at Institute AIFB at the Karlsruhe Institute of Technology (KIT) in Germany after pursuing a Ph.D. at the Digital Enterprise Research Institute (DERI) at the National University of Ireland, Galway. His research interests are large-scale data interoperation on the Semantic Web, Linked Data, knowledge representation, computational logic and user interaction on web data. He has published several dozen papers in these areas, and is author of several open source software systems. Two of his systems were awarded prizes at the Semantic Web Challenge co-located with the International Semantic Web Conference. In 2012, he received the ESWC 7-Years Most Influential Paper Award.

Dr. Harth has participated in numerous EU and national projects, participated in various program committees, and has served in the W3C Semantic Web Best Practices and Deployment and Rules Interchange Format working groups. In addition, he served as program committee member of numerous

[19] That is, interactions between autonomous agents and their hypermedia-based environments (cf. RQ2), but also interactions among autonomous agents and people on the Web (cf. RQ3).

[20] https://hyperagents.org

conferences and is one of the co-organizers of the Consuming Linked Data (COLD) workshop series and of the Semantic Web Challenge. He co-organized the Dagstuhl seminar 13252 "Interoperation in Complex Information Ecosystems" and is general chair of the Extended Semantic Web Conference ESWC in 2020.

D. Alessandro Ricci

Prof. Dr. Alessandro Ricci is an associate professor of the Department of Computer Science and Engineering, University of Bologna (Italy). His research interests concern agents and multi-agent systems as a paradigm for modeling, designing and programming software systems. His main research contributions are in the context of MAS programming and Agent-Oriented Software Engineering. These include: the "artifact" abstraction in agent and the Agents and Artifacts (A&A) conceptual model; the CArtAgO platform for environment programming; the JaCaMo platform, integrating CArtAgO with Jason and MOISE. Dr. A. Ricci is serving on the program committees of reference conferences for Agents and MAS such as AAMAS, IJCAI, AAAI and workshops. He served also as co-organizer of relevant initiatives in the Agents and Multi-Agent Systems community, such as the Engineering Multi-Agent Systems (EMAS) workshop, Programming Multi-Agent Systems (ProMAS) workshop. In the context of the SPLASH Conference (i.e. the reference International Conference on Software Development, hosting e.g. OOPSLA), in 2011 he started the organisation of a workshop called AGERE! about Programming paradigms based on Actors and Agents - in cooperation with main names of the Actor community, such as Gul Agha and Akinori Yonezawa. The workshop is still running (in 2019) inside SPLASH. Dr. A. Ricci has presented tutorials on Multi-Agent Oriented Programming at various edition of EASSS, the European Agent Systems Summer School (in 2009, 2010, 2011, 2012), and AAMAS (in 2015).

References

[1] M. Amundsen. *RESTful Web Clients: Enabling Reuse Through Hypermedia*. O'Reilly Media, 2017.

[2] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, 2001.

[3] C. Bormann, M. Ersue, and A. Keranen. Terminology for Constrained-Node Networks. RFC 7228 (Informational), May 2014.

[4] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1):107 – 117, 1998. Proceedings of the Seventh International World Wide Web Conference.

[5] C. Buil-Aranda, A. Hogan, J. Umbrich, and P.-Y. Vandenbussche. SPARQL Web-Querying Infrastructure: Ready for Action? In H. Alani, L. Kagal, A. Fokoue, P. Groth, C. Biemann, J. X. Parreira, L. Aroyo, N. Noy, C. Welty, and K. Janowicz, editors, *The Semantic Web – ISWC 2013*, pages 277–293, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[6] S. Capadisli and A. Guy. Linked Data Notifications, W3C Recommendation 2 May 2017. W3C Recommendation, World Wide Web Consortium (W3C), May 2 2017.

[7] R. Chinnici, J.-J. Moreau, A. Ryman, and S. Weerawarana. Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language, W3C Recommendation 26 June 2007. W3C Recommendation, World Wide Web Consortium (W3C), Jun. 26 2007.

[8] A. Ciortea, O. Boissier, and A. Ricci. Engineering world-wide multi-agent systems with hypermedia. In D. Weyns, V. Mascardi, and A. Ricci, editors, *Engineering Multi-Agent Systems*, pages 285–301, Cham, 2019. Springer International Publishing.

[9] A. Ciortea, O. Boissier, A. Zimmermann, and A. M. Florea. Responsive Decentralized Composition of Service Mashups for the Internet of Things. In *Proc. 6th International Conference on the Internet of Things*, IoT'16, pages 53–61, New York, NY, USA, 2016. ACM.

[10] A. Ciortea, S. Mayer, F. Gandon, O. Boissier, A. Ricci, and A. Zimmermann. A decade in hindsight: The missing bridge between multi-agent systems and the world wide web. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '19, pages 1659–1663, Richland, SC, 2019. International Foundation for Autonomous Agents and Multiagent Systems.

[11] A. Ciortea, S. Mayer, and F. Michahelles. Repurposing Manufacturing Lines on the Fly with Multi-agent Systems for the Web of Things. In *Proc. 17th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '18, pages 813–822, Richland, SC, 2018.

[12] O. Corby, C. Faron-Zucker, and F. Gandon. Ldscript: A linked data script language. In C. d'Amato, M. Fernandez, V. Tamma, F. Lecue, P. Cudré-Mauroux, J. Sequeda, C. Lange, and J. Heflin, editors, *The Semantic Web – ISWC 2017*, pages 208–224, Cham, 2017. Springer International Publishing.

[13] O. Dikenelli. Seagent mas platform development environment. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems: Demo Papers*, AAMAS '08, pages 1671–1672, Richland, SC, 2008. International Foundation for Autonomous Agents and Multiagent Systems.

[14] O. Dikenelli, O. Alatlı, and R. C. Erdur. Where are all the semantic web agents: Establishing links between agent and linked data web through environment abstraction. In D. Weyns and F. Michel, editors, *Agent Environments for Multi-Agent Systems IV*, pages 41–51. Springer International Publishing, Cham, 2015.

[15] J. Exposito, J. Ametller, and S. Robles. Configuring the JADE HTTP MTP. http://jade.tilab.com/documentation/tutorials-guides/configuring-the-jade-http-mtp/, 2010. Accessed: 20.04.2018.

[16] R. T. Fielding. *Architectural styles and the design of network-based software architectures*. PhD thesis, University of California, Irvine, 2000.

[17] R. T. Fielding and R. N. Taylor. Principled Design of the Modern Web Architecture. *ACM Trans. Internet Technol.*, 2(2):115–150, May 2002.

[18] R. E. Fikes and N. J. Nilsson. Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3):189 – 208, 1971.

[19] Foundation for Intelligent Physical Agents. FIPA Agent Message Transport Protocol for HTTP Specification. http://www.fipa.org/specs/fipa00084/SC00084F.html, 2002. Document number: SC00084F.

[20] F. Gandon. A Survey of the First 20 Years of Research on Semantic Web and Linked Data. *Revue des Sciences et Technologies de l'Information - Série ISI : Ingénierie des Systèmes d'Information*, Dec. 2018.

[21] M. P. Georgeff and A. L. Lansky. Reactive reasoning and planning. In *Proceedings of the Sixth National Conference on Artificial Intelligence - Volume 2*, AAAI'87, pages 677–682. AAAI Press, 1987.

[22] A. Gouaïch and M. Bergeret. Rest-a: An agent virtual machine based on rest framework. In Y. Demazeau, F. Dignum, J. M. Corchado, and J. B. Pérez, editors, *Advances in Practical Applications of Agents and Multiagent Systems*, pages 103–112. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[23] M. E. Gregori, J. P. Cámara, and G. A. Bada. A jabber-based multi-agent system platform. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '06, pages 1282–1284, New York, NY, USA, 2006. ACM.

[24] D. Guinard, V. Trifa, F. Mattern, and E. Wilde. *From the Internet of Things to the Web of Things: Resource-oriented Architecture and Best Practices*, pages 97–129. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[25] M. J. Hadley. Web Application Description Language (WADL). Technical report, Mountain View, CA, USA, 2006.

[26] S. Harris and A. Seaborne. SPARQL 1.1 Query Language, W3C Recommendation 21 March 2013. W3C Recommendation, World Wide Web Consortium (W3C), Mar. 21 2013.

[27] J. Hendler. Where are all the intelligent agents? *IEEE Intelligent Systems*, 22(3):2–3, May 2007.

[28] J. Hendler and A. Mulvehill. *Social Machines: The Coming Collision of Artificial Intelligence, Social Networking, and Humanity*. Apress, Berkely, CA, USA, 1st edition, 2016.

[29] N. R. Jennings and M. Wooldridge. *Applications of Intelligent Agents*, pages 3–28. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[30] S. Kaebisch, T. Kamiya, M. McCool, V. Charpenay, and M. Kovatsch. Web of Things (WoT) Thing Description, W3C Recommendation 9 April 2020. W3C Recommendation, World Wide Web Consortium (W3C), Apr. 9 2020.

[31] T. Käfer and A. Harth. Rule-based programming of user agents for linked data. In *Proceedings of the 11th International Workshop on Linked Data on the Web (LDOW) at the 27th Web Conference (WWW)*, 2018.

[32] J. Kopecký, T. Vitvar, C. Bournez, and J. Farrell. SAWSDL: Semantic Annotations for WSDL and XML Schema. *IEEE Internet Computing*, 11(6):60–67, Nov 2007.

[33] M. Kovatsch, Y. N. Hassan, and S. Mayer. Practical semantics for the Internet of Things: Physical states, device mashups, and open questions. In *Proc. 5th International Conference on the Internet of Things*, pages 54–61, Oct 2015.

[34] M. Lanthaler and C. Gütl. Hydra: A vocabulary for hypermedia-driven web apis. In *Proceedings of the WWW2013 Workshop on Linked Data on the Web*, volume 996 of *CEUR WS*, 2013.

[35] J. Lathem, K. Gomadam, and A. P. Sheth. SA-REST and (S)mashups : Adding Semantics to RESTful Services. In *International Conference on Semantic Computing (ICSC 2007)*, pages 469–476, Sep. 2007.

[36] D. Martin, M. Paolucci, S. McIlraith, M. Burstein, D. McDermott, D. McGuinness, B. Parsia, T. Payne, M. Sabou, M. Solanki, N. Srinivasan, and K. Sycara. Bringing Semantics to Web Services: The OWL-S Approach. In J. Cardoso and A. Sheth, editors, *Semantic Web Services and Web Process Composition*, pages 26–42, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[37] S. Mayer, R. Verborgh, M. Kovatsch, and F. Mattern. Smart configuration of smart environments. *IEEE Transactions on Automation Science and Engineering*, 13(3):1247–1255, July 2016.

[38] N. Mitra and Y. Lafon. SOAP Version 1.2 Part 0: Primer (Second Edition), W3C Recommendation 27 April 2007. W3C Recommendation, World Wide Web Consortium (W3C), Apr. 27 2007.

[39] D. Mitrović, M. Ivanović, Z. Budimac, and M. Vidaković. Radigost: Interoperable web-based multi-agent platform. *Journal of Systems and Software*, 90:167 – 178, 2014.

[40] J. P. Müller and K. Fischer. Application impact of multi-agent systems and technologies: A survey. In O. Shehory and A. Sturm, editors, *Agent-Oriented Software Engineering: Reflections on Architectures, Methodologies, Languages, and Frameworks*, pages 27–53, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.

[41] S. Niederer and J. van Dijck. Wisdom of the crowd or technicity of content? wikipedia as a sociotechnical system. *New Media & Society*, 12(8):1368–1387, 2010.

[42] D. A. Norman. *Design of everyday things: Revised and expanded*. Basic Books, Inc., New York, NY, USA, 2013.

[43] C. Pautasso, O. Zimmermann, and F. Leymann. RESTful Web Services vs. "Big" Web Services: Making the Right Architectural Decision. In *Proceedings of the 17th Intl. Conference on World Wide Web*, WWW '08, pages 805–814, New York, NY, USA, 2008. ACM.

[44] A. Ricci, M. Piunti, and M. Viroli. Environment programming in multi-agent systems: an artifact-based perspective. *Autonomous Agents and Multi-Agent Systems*, 23(2):158–192, 2011.

[45] D. Roman, U. Keller, H. Lausen, J. De Bruijn, R. Lara, M. Stollberg, A. Polleres, C. Feier, C. Bussler, and D. Fensel. Web service modeling ontology. *Applied ontology*, 1(1):77–106, 2005.

[46] Z. Shelby, K. Hartke, and C. Bormann. The Constrained Application Protocol (CoAP). RFC 7252 (Proposed Standard), June 2014.

[47] Q. Z. Sheng, X. Qiao, A. V. Vasilakos, C. Szabo, S. Bourne, and X. Xu. Web services composition: A decade's overview. *Information Sciences*, 280:218 – 238, 2014.

[48] M. P. Singh and M. N. Huhns. *Service-Oriented Computing: Semantics, Processes, Agents*. John Wiley & Sons, 2006.

[49] S. Speicher, J. Arwe, and A. Malhotra. Linked Data Platform 1.0, W3C Recommendation 26 February 2015. W3C Recommendation, World Wide Web Consortium (W3C), Feb. 26 2015.

[50] S. Stadtmüller, S. Speiser, A. Harth, and R. Studer. Data-fu: A language and an interpreter for interaction with read/write linked data. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 1225–1236, New York, NY, USA, 2013. ACM.

[51] R. Verborgh, T. Kuhn, and T. Berners-Lee. *Proc. 2nd Workshop on Decentralizing the Semantic Web co-located with the 17th International Semantic Web Conference (ISWC 2018)*, volume 2165. 2018.

[52] R. Verborgh, T. Steiner, D. Van Deursen, S. Coppens, J. Gabarró Vallés, and R. Van de Walle. Functional descriptions as the bridge between hypermedia APIs and the Semantic Web. In *Proceedings of the Third International Workshop on RESTful Design*, pages 33–40. ACM, Apr. 2012.

[53] R. Verborgh, M. Vander Sande, O. Hartig, J. Van Herwegen, L. De Vocht, B. De Meester, G. Haesendonck, and P. Colpaert. Triple Pattern Fragments: a low-cost knowledge graph interface for the Web. *Journal of Web Semantics*, 37–38:184–206, Mar. 2016.

[54] G. Weiss. *Multiagent systems: A modern approach to distributed artificial intelligence*. MIT press, 2000.

[55] D. Weyns and F. Michel. Agent environments for multi-agent systems – a research roadmap. In D. Weyns and F. Michel, editors, *Agent Environments for Multi-Agent Systems IV*, pages 3–21. Springer International Publishing, Cham, 2015.

[56] D. Weyns, A. Omicini, and J. Odell. Environment as a first class abstraction in multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 14(1):5–30, Feb 2007.

[57] S. Willmott, J. Dale, B. Burg, P. Charlton, and P. O'Brien. Agentcities: A worldwide open agent network. *Agentlink News*, 8, 2001.