Report from Dagstuhl Seminar 23081

Abstractions for Agents on the Web

Edited by

Antoine Zimmermann[1], Andrei Ciortea[2,3], Catherine Faron[3], Jomi F. Hübner[4], Eoin O'Neill[5], María Poveda-Villalón[6], and Alessandro Ricci[7]

1 **Mines Saint-Étienne, Univ. Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, F-42023, Saint-Étienne, France** antoine.zimmermann@emse.fr
2 **University of St. Gallen, St. Gallen, Switzerland** andrei.ciortea@unisg.ch
3 **Université Côte d'Azur, CNRS, Inria, I3S, Sophia Antipolis, France** faron@i3s.unice.fr
4 **Department of Automation and Systems, Universidade Federal de Santa Catarina** jomi.hubner@ufsc.br
5 **University College Dublin, Dublin, Ireland** eoin.o-neill.3@ucdconnect.ie
6 **Ontology Engineering Group, Universidad Politécnica de Madrid, Madrid, Spain** mpoveda@fi.upm.es
7 **Alma Mater Studiorum - Università di Bologna, Cesena Campus, Italy** a.ricci@unibo.it

Abstract

This document reports on the findings of a Working Group at the Dagstuhl seminar on *Agents on the Web* relative to what abstractions must be considered when designing multi-agent systems on the Web. We first consider the case of a single agent that makes use of the Web, consuming or producing information. Then, we examine the case of multiple agents that work together as a system where the Web plays the central role of the environment in which they are situated. Finally, we provide preliminary ideas on how to instantiate the elicited abstractions in practice, relying on Semantic Web models and technologies and the Solid Protocol for Social Linked Data.

Seminar 19.–24. February, 2023 – https://www.dagstuhl.de/23081
2012 ACM Subject Classification Computing methodologies → Artificial intelligence; Computing methodologies → Multi-agent systems
Keywords and phrases Multi-agent systems, Web, norms
Digital Object Identifier 10.4230/DagRep.1.1.1

1 Introduction

Agents that are (inter)acting autonomously on the Web and with the Web must make use of the information available in this environment to gather knowledge and make decisions. Different kinds of information correspond to different types of abstractions that we want to formally identify such that they can be represented explicitly and systematically. With such representations of relevant abstractions, artificial agents can replicate activities on the Web that have been so far reserved to human agents: navigating hypermedia environments with a purpose, choosing what next links to follow based on promising signifiers, and collaborating with one another via collaborative Web platforms.

In what we consider here in the context of agents on the Web, an abstraction is a notion that can be defined as a formal structure (such as a mathematical object) which is relevant for a system of autonomous agents to function adequately. If the agents are humans, it is usually not necessary to formalise these abstractions because people have many ways of getting an intuitive understanding of ways to cooperate. For instance, when a community of people discusses issues on a Web forum, the fact that there is a coloured rectangle with the words "New thread" is sufficient to alert humans that this is a button for starting a new textual discussion. When one sees a blue underlined word that says “Reply” under the last message of a thread, one easily understands that this is a link that allows participants to reply to the thread, or to the last post. On the contrary, artificial agents may not be able to interpret these visual cues. If we would like to allow such agents to participate in a discussion, it is possible to hard code the interactions, e.g., a developer translates the actions needed to read and post messages on a specific forum. However, this is very limiting in terms of autonomy. Instead, we would like to define more abstractly what generic notions have to be known by the agent in order to choose its interactions with Web resources. As an example, online discussions can be characterised by the fact that there are discussion threads, each composed of individual messages that one can respond to. Messages are authored by someone (or something) at a date and time. These concepts are common to any Web forum, to blogs and microblogging platforms, to Git issues, chats, etc.

Once identified, the concepts that pertain to one type of abstractions can be formally defined as terms of an ontology, further described and constrained by logical axioms. In turn, a web platform can make use of such terms to describe itself for autonomous agents. The aim of this document is to report on the findings of a Working Group at the Dagstuhl seminar on *Agents on the Web* relative to what abstractions must be considered when designing multi-agent systems on the Web. Consequently, this can serve as a starting point to decide what ontologies may be used to describe where, what, how resources are to be used by autonomous agents, for themselves or for a collective goal.

To do this, we start by considering the case of a single agent that makes use of the Web, consuming or producing information. In this first case, the Web is only used as a tool that is assumed to evolve independently of all agents (Section 2). Then, we examine the case of multiple agents that work together as a system where the Web plays the central role of the environment in which they are situated (Section 3). Then, we provide preliminary ideas on how to instantiate the abstractions in practice (Section 4).

2 Single agents interacting with the Web

Before considering the more arduous challenges of coordinating multiple agents on and via the Web, we first tackle the case where a single agent makes use of the Web as if it was a large warehouse full of documents and tools. The agent may exploit the Web in getting information from it: from the Web of documents, the Web of data, or the Web of knowledge. From this, it can derive or update its own beliefs. Or the agent may act upon resources leading to altering the state of the Web itself (e.g., posting a new message adds text to a Web page), or altering the state of the physical world itself (e.g., moving a robotic arm that offers a Web interface using Web of Things technologies).

To be fully autonomous on these types of tasks, Web agents must start their exploration from somewhere. At this point, there must be indications of where to find relevant information. This should lead them to online platforms where they may ask themselves: *What can I do on this platform? How can I do it?* Since the Web is evolving, agents may also be interested in changes occurring, without constantly checking for differences. An agent can be partly or fully guided by hard-coded knowledge provided at implementation time, such as the URIs of a preselected set of resources. However, in general, the agent has to interact with an entry point that serves as a hub towards any relevant resources. Such hub can be a search engine, a query endpoint, etc. that provides, at the minimum, links to resources that match the agent's request. Usually, the resource that the agent is looking for is not atomic: it corresponds to a compound resource that gives multiple options to the agent. For instance, the agent may be looking for online forums. Each online forum is a resource that provides discussion threads, search functionalities, posting abilities, etc. We call the abstraction of such compound resource a *platform*. The functionalities enabled by a platform and that the agent can use are affordances, which can be indicated to the agent via signifiers [10].

In summary, some competency questions must be answerable using the right abstractions, as follows:

- *Where do I find relevant resources?* Entry hubs, links to platforms.
- *What can I do on this platform?* Signifiers, links to affordances.
- *How can I do it?* Action possibilities, links to Manual.
- *How can I know what changes?* Notifications.

3 Multiple agents interacting on the Web

In the case of a single agent interacting with the Web, all online resources are considered as artifacts or tools that can evolve according to an independent life cycle. When agents need to collaborate, cooperate, or at least behave with awareness of each other on the Web, additional abstractions need to be introduced.

The concept of situatedness with regards to Web agents in a *multiagent* context can be defined in relation to the scope of environment that they exist within. As defined in Section 2, any tools that an agent utilises in its goal-directed behaviour provides a level of situatedness as each interaction will be between two entities that can be identified using the global, unique naming scheme provided by the use of IRIs. This also pertains to agent-to-agent interaction as when agents collaborate and coordinate in order to achieve system-wide, or individual, goals they interact via the Web, with each agent being identifiable by a unique IRI.

In order for collaboration and coordination between Web agents to be a possibility in a multiagent context, the embodiment of an agent within a hypermedia environment is a crucial abstraction that needs to be defined. This allows for the agent to maintain a presence in a Web context which enables other agents to be aware of its existence. However, agents can be embodied in multiple Web contexts, the same way a human would have multiple profiles on different social media platforms. This embodiment of an agent in a particular Web context can represent an identity of a Web agent. The agent's profile can be a certain persona that this particular agent wants to portray within that context. This abstraction can contain the agent's preferred methods of interaction via its *communication interface*, which can be defined as or subject to a set of *norms*. These *norms* can be defined within the hypermedia environment, as resources, that can be directly associated with the Web agent's embodiment.

When discussing multiple Web agents in a hypermedia environment, the scope of the environment is a relevant topic due to the computational limitations of the agents themselves. As a result, an abstraction for specific collections of resources can be viewed as an aggregation of related resources. This abstraction can be viewed as an *area* within a Web-based environment. If we take for example a set of related resources that represents elements of a building

(Rooms, Floors etc.), these can be aggregated into an *area*. This example makes logical sense as the relation can be easily visualised and the relations envisioned, however there can be Web resources that are related, but in a less explicit manner and so this abstraction can provide a general way of aggregating resources to provide agents with scope as to the environments they inhabit.

4 Instantiating the abstractions

The Semantic Web relies on the Resource Description Framework (RDF), a graph model to structure data by expressing relations between entities, and on RDF Schema and the Web Ontology Language to represent the ontologies used in RDF knowledge graphs, thus providing semantics to them. The Linked Data principles are a set of best practices to publish RDF data on the Web, namely: 1) use URIs to name things; 2) use HTTP URIs so that names can be looked up; 3) describe things using standards (RDF) so useful information is provided for URIs; and 4) include links to other URIs in things descriptions. Ontologies and linked data together provide the means by which an agent can reliably interpret resources described on the Web, whether they are digital resources or real-world resources. Additionally, with links, a web resource leads to other resources, and so forth, so as to make agents aware of the environment that the Web constitutes.

Besides, the Solid Protocol [2] aims at decentralising personal data management in such a way that Web users regain ownership and control over their data. At the core of it, there is the Solid *pod* (**p**ersonal **o**nline **d**ata store) that hosts the user's data and is implemented as a Linked Data Platform [9] with access control on top of it. Solid pods can host any kind of data but are designed in particular to easily manage RDF datasets with fine-grained read/write operations. Overall, the Solid Protocol specifies authentication, storage, access control, and interactions that must be implemented by Solid pods and Solid platforms in order to interoperate with each other and with applications that build on them.

Solid pods containing RDF knowledge graphs can be used to implement the abstractions introduced in Section 3. In order to represent the different levels and types of knowledge defining those abstractions it is necessary to combine a number of ontologies. For example the Web of Things model [7] might be used to describe the affordances while the Open Digital Rights Language (ODRL [6]) would be useful to define access policies to data or resources. Other existing ontologies might be extended or specialised to represent particular needs in the agents domain like the Organization ontology [8] or sensor and actuator related ontologies as SSN&SOSA [5] and SAREF [4]. Finally, specific domain ontologies related to the agent's tasks should be develop or reused. The Solid protocol states that "an agent is a person, social entity, or software identified by a URI; e.g., a WebID denotes an agent". We then assume that such a URI would dereference to an entry point for the data pod of the agent, where an **Agent Description** would be provided as an RDF graph, in addition to the mandatory credentials for authenticating the agent. We call the Solid pod implementing an agent's Web body a *pody* [12].

5 Conclusion

The dominant view in AI research is that intelligence is defined in relation to the environment an agent occupies [11]. The agents we have on the Web today also reflect the nature of their environment: they mostly solve the problem of finding and curating information in a Web of Documents. The Web, however, was designed to provide different levels of abstraction—going from computer networks to a knowledge-level representation of the world [1]. With recent standardisation efforts for Linked Data and subsequently the Web of Thing, the Web now extends to the physical world and provides agents with a uniform, knowledge-level hypermedia fabric that allows them "to browse and manipulate reality"—a vision that can be traced back to the early days of the Web.[1] This evolution unlocks new practical use cases for more intelligent agents on the Web, but such agents have to be provided with a proper level of abstraction that allows them to discover and interact with one another—and with the world in general. This report presents an initial proposal for such a set of abstractions with a focus on *situatedness* and *embodiment*. We invite the research community to join us in further investigating and developing these abstractions.

Acknowledgements

Andrei Ciortea and Antoine Zimmermann were supported by funding from the HyperAgents project (grant from French national research agency ANR-19-CE23-0030-01 and from Swiss National Science Foundation No. 189474). María Poveda-Villalón was supported by the Spanish National project KnowledgeSpaces: Técnicas y herramientas para la gestión de grafos de conocimientos para dar soporte a espacios de datos (PID2020-118274RB-I00). Eoin O'Neill is supported by the Science Foundation of Ireland Strategic Partnerships Programme (16/SPP/3296) and is co-funded by Origin Enterprises Plc.

References

1 Tim Berners-Lee. Levels of Abstraction: Net, Web, Graph. Design Issues, 2007. `https://www.w3.org/DesignIssues/Abstractions.html`

2 Sarven Capadisli, Tim Berners-Lee, Ruben Verborgh and Kjetil Kjernsmo. Solid Protocol, W3C Solid Community Group working draft, 2022. `https://solidproject.org/TR/2021/protocol-20211217`

3 Sarven Capadisli and Amy Guy. Linked Data Notification. W3C Recommendation, 2 May 2017. `https://www.w3.org/TR/2017/REC-ldn-20170502/`

4 Laura Daniele, Raúl García-Castro, Maxime Lefrançois and María Poveda-Villalón. SAREF: the Smart Applications REFerence ontology. ETSI TS 103 264, v3.1.1, 2020. `https://saref.etsi.org/core/v3.1.1/`

5 Armin Haller, Krzysztof Janowicz, Simon Cox, Danh Le Phuoc, Jamie Taylor and Maxime Lefrançois. Semantic Sensor Network Ontology, W3C Recommendation, 2017. `https://www.w3.org/TR/2017/REC-vocab-ssn-20171019/`

6 Renato Ianella and Serena Villata. ODRL Information Model 2.2, W3C Recommendation, 2018 `https://www.w3.org/TR/2018/REC-odrl-model-20180215/`

7 Sebastian Kaebisch, Michael McCool and Ege Korkan. Web of Things (WoT) Thing Description 1.1, W3C Candidate Recommendation Snapshot, 2023. `https://www.w3.org/TR/2023/CR-wot-thing-description11-20230119/`

8 Dave Reynolds. The Organization Ontology, W3C Recommendation, 2014. `http://www.w3.org/TR/2014/REC-vocab-org-20140116/`

[1] See the keynote of Sir Tim Berners-Lee at the First International Conference on the World Wide Web (WWW'94): `https://videos.cern.ch/record/2671957`

9 Steve Speicher, John Arwe and Ashok Malhotra. Linked Data Platform 1.0, W3C Recommendation 2015. https://www.w3.org/TR/2015/REC-ldp-20150226/

10 Danai Vachtsevanou, Andrei Ciortea, Simon Mayer, and Jérémy Lemée. Signifiers as a First-Class Abstraction in Hypermedia Multi-Agent Systems. In Proc. of the 22nd International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2023.

11 Michael Wooldridge. Intelligent Agents. In *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, MIT Press, 2000. p27–72.

12 Antoine Zimmermann, Andrei Ciortea, Catherine Faron, Eoin O'Neill and María Poveda-Villalón. Pody: a Solid-based Approach to Embody Agents in Web-based MAS. To appear in Proc. of Engineering Multiagent Systems, EMAS 2023.