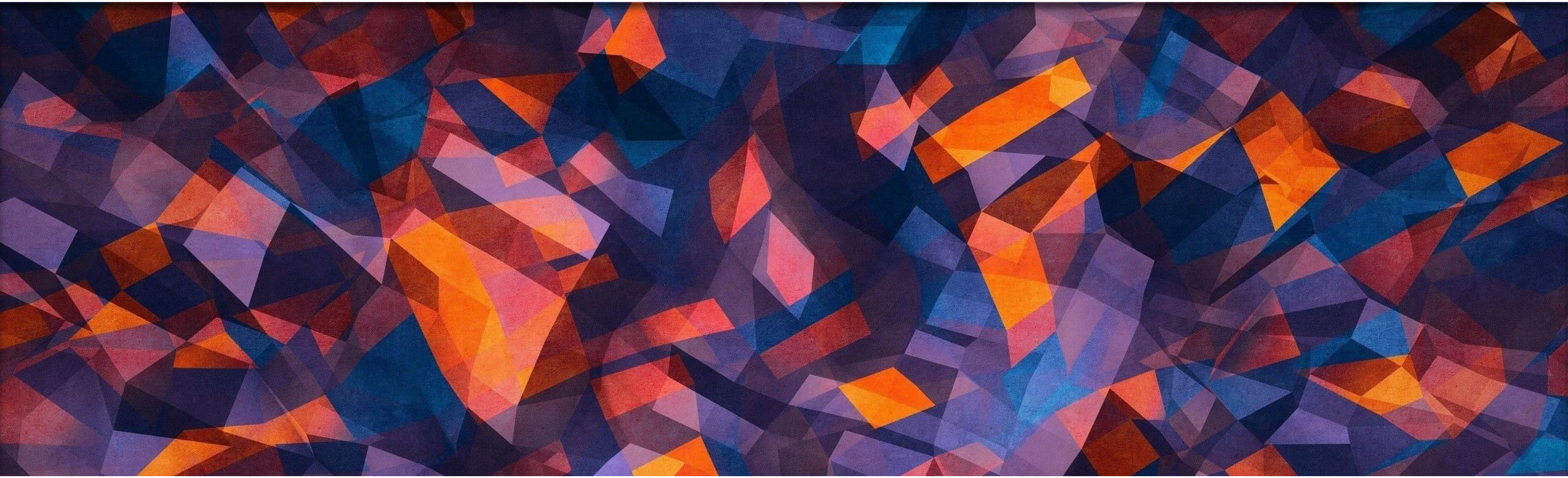
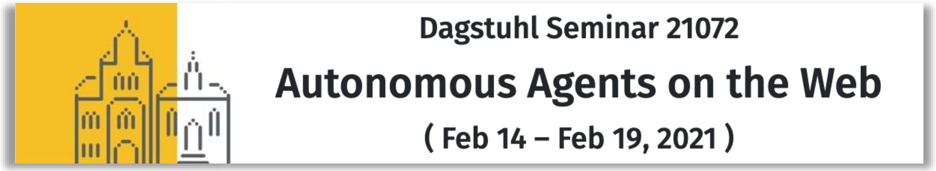


October 25, 2025 | Bologna & Online

W3C WebAgents Community Group

Andrei Ciortea, Rem Collier, Ege Korkan, and Antoine Zimmermann

Agents on the Web: Community



https://www.dagstuhl.de/21072

May 2019 — HyperAgents 2019 @ TheWebConf 2019

Feb. 2021 — Dagstuhl Seminar 21072: “Autonomous Agents on the Web”

February 2023 — Dagstuhl Seminar 23081: “Agents on the Web”

https://www.dagstuhl.de/23081

March 2023 — W3C WebAgents Community Group

https://www.w3.org/community/webagents/

October 2025 — HyperAgents 2025 @ ECAI 2025

HyperAgents 2025

Today's Focus: Architectural Patterns

W3C Community Group Draft Report

TABLE OF CONTENTS

6 ReSpec

WebAgents Community Group Report on Interoperability for Agents on the Web

Draft Community Group Report 25 October 2025

Latest published version:
none

Latest editor's draft:
https://w3c-cg.github.io/webagents/TaskForces/Interoperability/Reports/report-interoperability.html

Editors:
Andrei Ciortea (Inria and University of St.Gallen)
Rem Collier (University College Dublin)

Authors:
Jérémy Lemée (University of St.Gallen)
Your Name

Feedback:
GitHub w3c-cg/webagents (pull requests, new is

Abstract

Advances in large language models (LLMs) that can
autonomous agents and multi-agent systems. Like p
designed for specific tasks, highlighting the need for

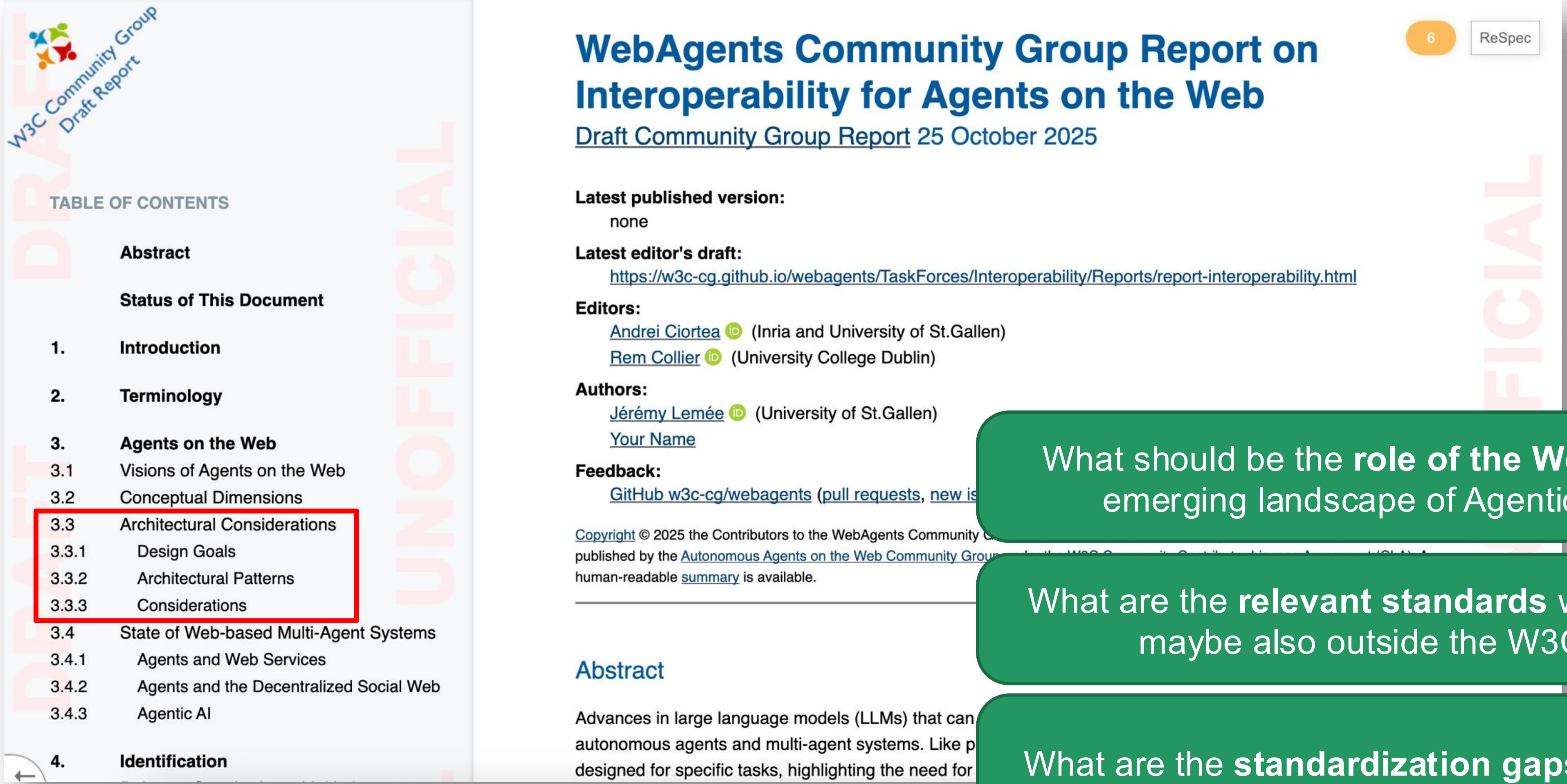
https://bit.ly/hyperagents2025-webagents-report

What should be the **role of the Web** in the emerging landscape of Agentic AI?

What are the **relevant standards** within and maybe also outside the W3C?

What are the **standardization gaps** (if any)?

Today’s Focus: Architectural Patterns



W3C Community Group Draft Report

DRAFT UNOFFICIAL

TABLE OF CONTENTS

6 ReSpec

WebAgents Community Group Report on Interoperability for Agents on the Web

Draft Community Group Report 25 October 2025

Latest published version:
none

Latest editor's draft:
https://w3c-cg.github.io/webagents/TaskForces/Interoperability/Reports/report-interoperability.html

Editors:
Andrei Ciortea (Inria and University of St.Gallen)
Rem Collier (University College Dublin)

Authors:
Jérémy Lemée (University of St.Gallen)
Your Name

Feedback:
GitHub w3c-cg/webagents (pull requests, new is

Abstract

Advances in large language models (LLMs) that can
autonomous agents and multi-agent systems. Like p
designed for specific tasks, highlighting the need for

https://bit.ly/hyperagents2025-webagents-report

What should be the **role of the Web** in the emerging landscape of Agentic AI?

What are the **relevant standards** within and maybe also outside the W3C?

What are the **standardization gaps** (if any)?

Today’s Focus: Architectural Patterns

Decentralized Identifiers (DIDs) v1.0

Core architecture, data model, and representations

W3C Recommendation 19 July 2022

Goal	Description
Decentralization	Eliminate the requirement for centralized authorities or single point failure in identifier management, including the registration of globally unique identifiers, public verification keys, services, and other information.
Control	Give entities, both human and non-human, the power to directly control their digital identifiers without the need to rely on external authorities.
Privacy	Enable entities to control the privacy of their information, including minimal, selective, and progressive disclosure of attributes or other data.
Security	Enable sufficient security for requesting parties to depend on DID documents for their required level of assurance.
Proof-based	Enable DID controllers to provide cryptographic proof when interacting with other entities.
Discoverability	Make it possible for entities to discover DIDs for other entities, to learn more about or interact with those entities.
Interoperability	Use interoperable standards so DID infrastructure can make use of existing tools and software libraries designed for interoperability.
Portability	Be system- and network-independent and enable entities to use their digital identifiers with any system that supports DIDs and DID methods.
Simplicity	Favor a reduced set of simple features to make the technology easier to understand, implement, and deploy.
Extensibility	Where possible, enable extensibility provided it does not greatly hinder interoperability, portability, or simplicity.

Today's Focus: Architectural Patterns

Decentralized Identifiers (DIDs) v1.0

Core architecture, data model, and representations

W3C Recommendation 19 July 2022

§ 1.3 Architecture Overview

This section is non-normative.

This section provides a basic overview of the major components of Decentralized Identifier architecture.

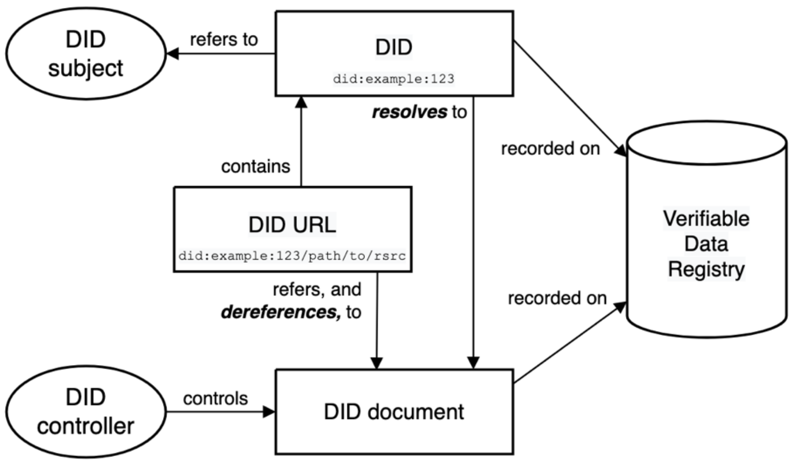


Figure 2 Overview of DID architecture and the relationship of the basic components. See also: narrative description.

Today's Agenda

CET	Agenda
14:00 - 14:15	Welcome & Agenda Setting
14:15 - 15:30	Focused Discussion
15:30 - 16:00	Coffee Break
16:00 - 17:00	Demo / Hands-on session
17:00 - 17:30	Wrap-up and Next Steps

What **design goals should guide** the design of Web-based Multi-Agent Systems (MAS)?

How can the Web contribute to those design goals?

What is a **minimal set of architectural patterns** for Web-based MAS?

What **architectural constraints** do we need in order to fully leverage the Web?

Design Goals / Non-functional Properties

MAS is a class of systems commonly known for [Weyns, 2010]:

- **adaptability**: the ability of agents to adapt to changes in their environment, including other agents entering or leaving the system, changes in interaction protocols enacted by agents, etc.
- **robustness**: the ability of a MAS to continue operating despite agent failures, often achieved by replicating capabilities across agents to ensure redundancy [Shehory, 98]
- **openness**: the ability of introducing additional agents into the system in excess to the agents that comprise it initially [Shehory, 98]:
  - ***off-line openness***: agents can only be added to the system at design time; the system has to be halted, adapted, and re-started
  - ***static openness***: agents can enter or leave the system without re-starting it, but either all other agents have to be notified or they all hold in advance a list of prospective agents
  - ***dynamic openness***: agents can enter or leave the system dynamically, at run time, without having to notify the other agents in the system
- **scalability**: the ability of MAS to accommodate a large number of agents or to deal with large amounts of data

Others:

- **transparency**: TBD
- **accountability**: TBD

O.M. Shehory: Architectural properties of multi-agent systems, 1998.
D. Weyns: Architecture-Based Design of Multi-Agent Systems. Springer, Berlin, Heidelberg, 2010.

Design Goals / Non-functional Properties

What the Web can offer:

- **scalability** (adapted from [van Steen & Tannenbaum, 2023]):
  - ***size scalability***: the ability of a system to grow in size (e.g., number of users, nodes, services) without a noticeable loss of performance
  - ***geographical scalability***: the ability of a system to spread over long distances without noticeable communication delays
  - ***administrative scalability***: the ability of a system to spread across many organization
- **heterogeneity**: the ability of a software system to consist of multiple disparate constituents or function in multiple disparate computing environments [Taylor et al., 2010]
- **evolvability**: the degree to which the implementation of a system component can be changed without negatively impacting other components [Fielding, 2000]
- **extensibility**: the ability to add functionality to a system [Fielding, 2000]

M. van Steen and A.S. Tanenbaum. Distributed Systems (Fourth Edition), 2023.
R.T. Fielding: Architectural Styles and the Design of Network-based Software Architectures. Ph.D. Thesis, UC Irvine, 2000.
R.N. Taylor, N. Medvidovic, and E.M. Dashofy. Software Architecture: Foundations, Theory, and Practice. John Wiley & Sons, Inc. 2010.

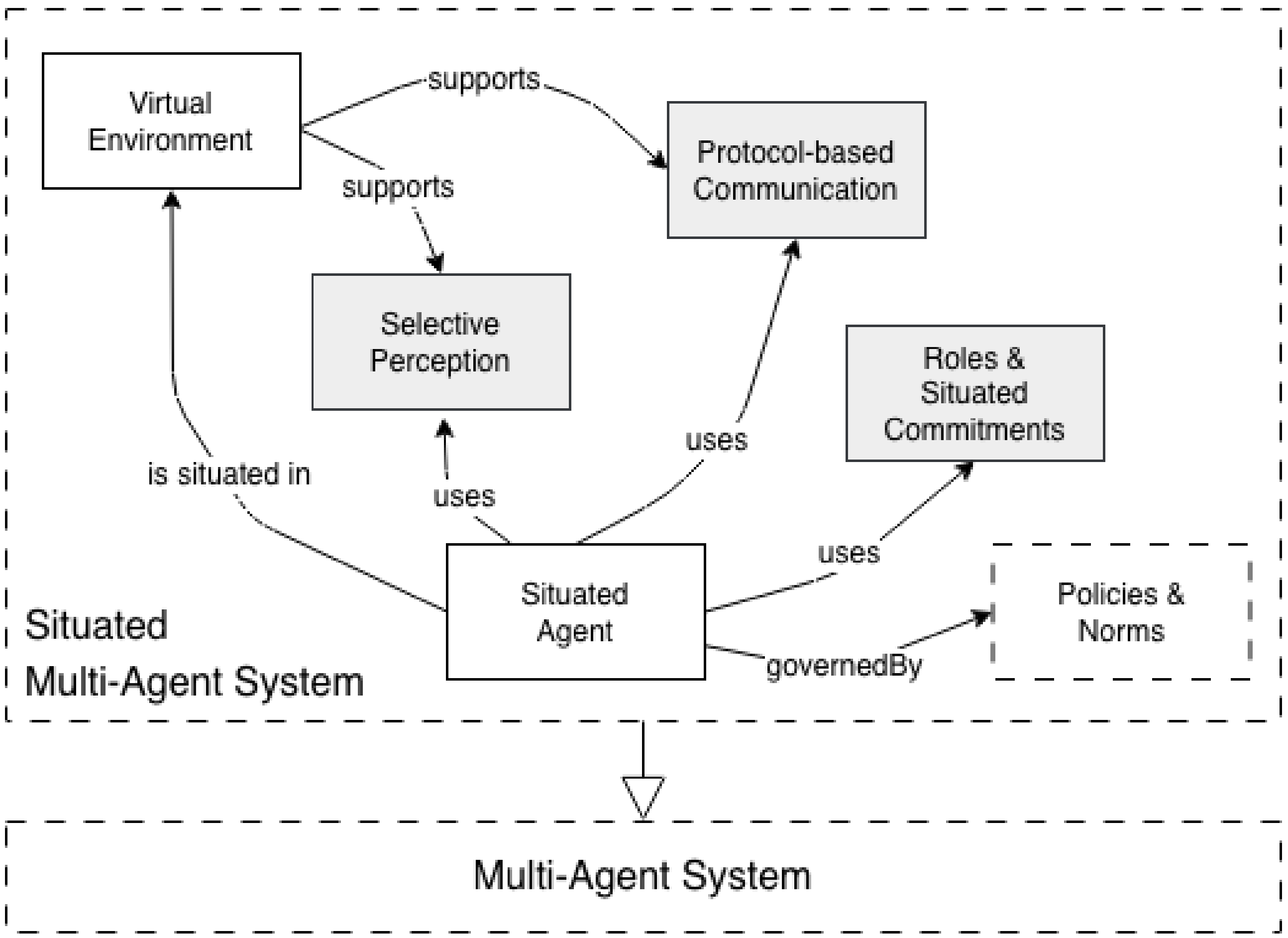
Design Goals / Non-functional Properties

Hewitt's requirements for open systems [Hewitt, 1990]:

- ***continuous change and evolution***: “Systems must be able to change as the components and demands placed upon them change. (…) Moreover, they must be able to evolve new internal components in order to accommodate the shifting work they perform.”
- ***arm's-length relationships and decentralized decision making***: “The computers, people, and agencies that make up open systems do not have direct access to one another's internal information. (…) This leads to decentralized decision making.”
- ***accommodating perpetual inconsistency among knowledge bases***: “Because of privacy and discretionary concerns, different knowledge bases will contain different perspectives and conflicting beliefs. (…) Decentralization makes it impossible to update all knowledge bases simultaneously.”
- ***the need for negotiation among system components***: “In a highly distributed system, no system component directly controls the resources of another. The various components of the system must persuade one another to provide capabilities.”
- ***recognition of the inadequacy of the closed-world assumption***: “Systems that depend on the closed-world assumption make use of the principle that they can find all existing instances of a concept by searching their local storage. At first glance it might seem that the closed-world assumption, almost universal in the AI literature, is smart because it provides a ready default answer for any query. Unfortunately, the default answers provided become less realistic as open systems increase in size and less of the information is available locally.”

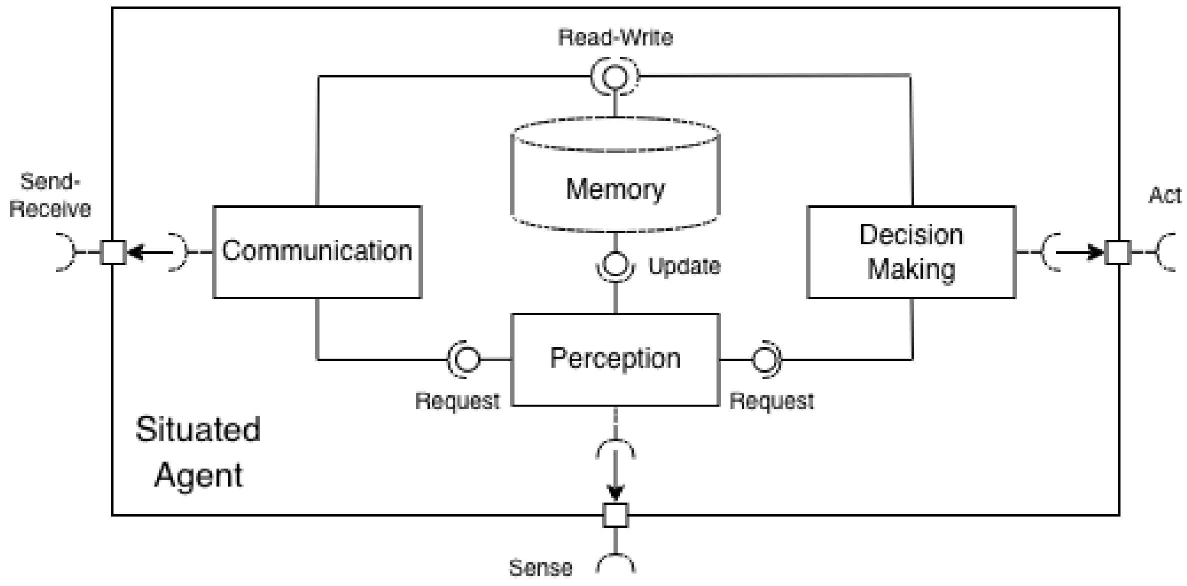
Carl Hewitt. The challenge of open systems. The Foundations of Artificial Intelligence, pages 383–395. Cambridge University Press, 1990.

Architectural Patterns for Situated MAS

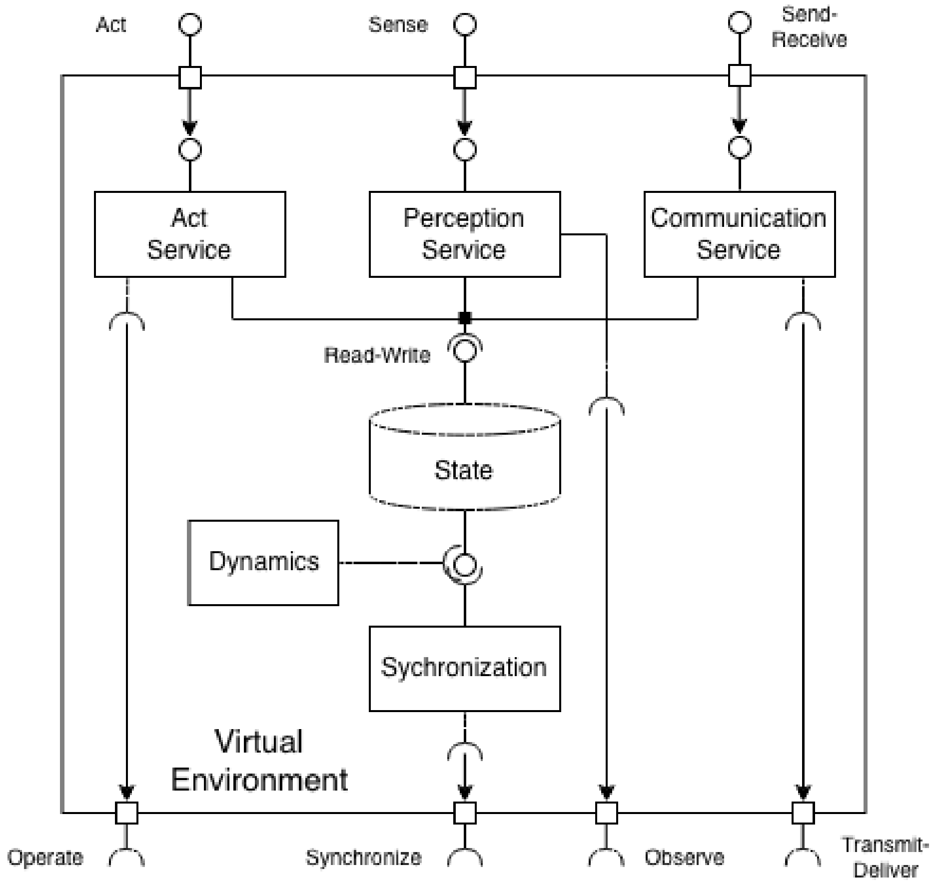


Overview of the Pattern Language

Architectural Patterns for Situated MAS



Situated Hypermedia Agent



Virtual Hypermedia Environment

Any Questions / Comments / Doubts / Concerns?

Images

https://freepik.com