

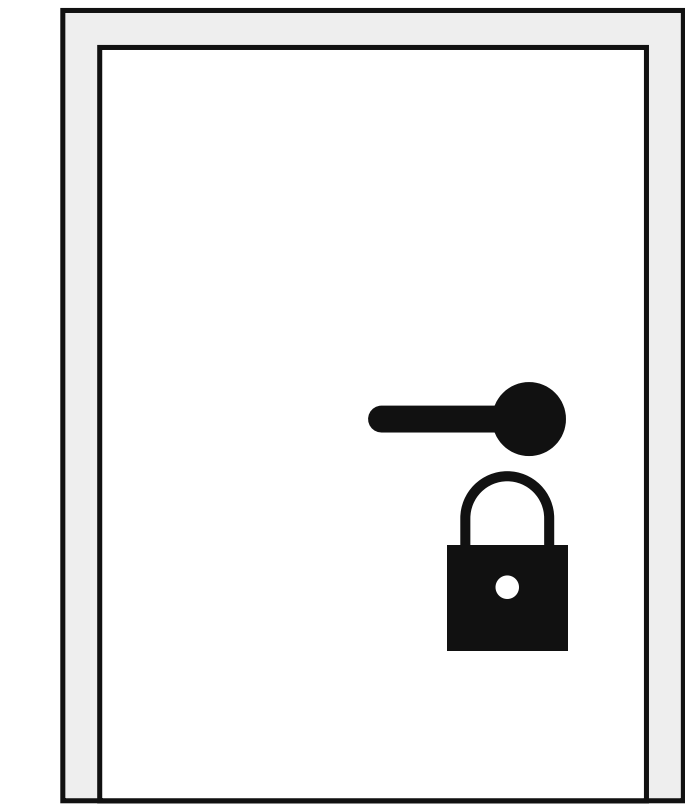
Hypermedia as a Substrate for the Agentic Web

From Documents to Affordances

Juho Vepsäläinen¹ · Fernando Miguel Carvalho²

¹ Department of Computer Science, Aalto University, Espoo, Finland

² Instituto Superior de Engenharia de Lisboa, Instituto Politécnico de Lisboa, Lisbon, Portugal



What can an agent do here, now?

A door handle makes an **affordance** visible: a possible action, opening the door. A lock reminds us that recognizing the action does not establish its current availability.

Web applications should expose currently valid, state-dependent actions alongside represented state. People and agents can inspect the same controls.

RESEARCH QUESTION

How should web applications expose currently valid, state-dependent actions for agentic use?

01 Follow the available action

The paper’s order-cancellation example turns this proposal into an inspectable contract. Controls are generated for the current resource state, request context and actor authorization.

INSPECT	IF SUCCESSFUL	RECOVER IF STALE
/orders/42 Paid Reason Duplicate ▾ Cancel order orderRevision: 7	/orders/42 Cancelled No cancellation control	Old representation 409 Conflict Return current representation

Paid order 42 exposes a cancellation form: a target, POST method, required reason, revision token, expected outcome and recovery meaning.

The server revalidates state and authorization. Success returns a cancelled representation without the cancellation control.

A representation can become stale. An old request receives 409 Conflict; the proposed response supplies current state and valid recovery actions.

The cancellation control in HTML	Meaning alongside the request
<pre><form action="/orders/42/cancellations" method="post"> <input type="hidden" name="orderRevision" value="7"> <label>Reason <select name="reason" required> <option>Duplicate</option><option>Mistake</option> </select> </label> <button>Cancel order</button> </form></pre> Adapted from Figure 2. HTML form excerpt at left; selected RDFa annotations at right. The affordance vocabulary is illustrative and non-standard.	The paper adds RDFa to identify the action, expected outcome and recovery behavior: <pre><link property="relation" href="/rels/cancel"> <meta property="outcome" content="cancelled"> <meta property="recovery" content="return-current"></pre>

02 Keep one shared model

Derive current-state controls and complementary interfaces from the same transition, validation and authorization model.

Transitions · Validation · Authorization
HTML: executable links, forms and input constraints.
MCP / OpenAPI: typed tools, operations, schemas and integration.
RDFa / JSON-LD: semantic enrichment. The example's affordance vocabulary is illustrative and non-standard.

03 Test the position

1. Conformance: one unchanged client, two independent stateful implementations. Expose exactly valid actions; reject invalid and stale requests.
2. Agent use: compare browser interaction, in-band controls and derived MCP under fixed model/task budgets. Measure valid completion and stale-state recovery.
3. Evolution: compare derived and independently maintained MCP contracts. Measure defects and authoring/governance effort.

WHAT REMAINS UNPROVEN	This is an architectural position, not evidence of improved agent performance. Richer representations, vocabulary governance and conformance tests add costs. Other interfaces may be primary for machine-only, batch or cross-system use.
-----------------------	--

When adding agents to your application, where will they discover valid actions?

Juho Vepsäläinen · juho.vepsalainen@aalto.fi