

A Foundational Note on Agents as Operational Semantics

Hsin-Yi Chen
SlashLife AI

Abstract

This note presents a minimal formulation in which an agent is the fixed point of a semantic operational law acting on the task structure and its graft. Task structure specifies the abstract organization of activity; graft provides the pragmatic linguistic action required for situated use. Identity, lifecycle, narrative continuity, and attestation arise as higher constructors of this fixed point. The formulation is substrate-agnostic and independent of any particular computational or representational architecture.

1 Premise

Classical approaches treat an agent as a predefined entity endowed with cognitive or computational attributes. In systems mediated by natural language and contextual action, such an approach obscures the generative process from which agency arises.

A more economical formulation is possible: agency emerges as the homotopy-stable semantic continuation generated by the interaction of abstract task structure with contextual graft.

2 Task Structure and Graft

Let T denote the *task structure*—the abstract, decomposable organization of a goal, workflow, or obligation. Sensory updating requires no separate treatment: admissible perceptual signals appear as morphisms into T .

Let G denote the *graft*—the pragmatic action that binds a task to its situated context. Graft encompasses deixis, presupposition, discourse role, and other operations ordinarily realized through natural language. It is modality-agnostic: textual, visual, auditory, or behavioural realizations are all treated as semantic representations inhabiting different image spaces of the same core.

Both T and G concern semantic form rather than surface expression.

3 Agents as Fixed Points of a Semantic Operational Law

Here “operational semantics” is used in the semantic rather than computational sense: a law-like operator on semantic states, independent of evaluation strategy or machine model.

I adopt the following definition:

$$\text{Agent} := \mu X. \text{OpSem}(T \times G \rightarrow X). \quad (1)$$

Equation (1) asserts:

- each task–graft pair induces a semantic transformation on a state X ;
- the agent is the minimal fixed point of this transformation;
- agency is a self-stabilizing semantic continuation rather than an ontological object.

No commitment is made to Turing machines, symbolic architectures, neural inference systems, or hybrid models; the definition is representation- and substrate-independent.

4 Higher-Inductive Structure

The fixed-point formulation supports derived higher structure.

Identity. Identity corresponds to a path in the fixed point:

$$a = a' : \equiv \text{path in } \mu X. \text{OpSem}(T \times G \rightarrow X).$$

Lifecycle. Lifecycle is the inductive unfolding:

$$a_0 \xrightarrow{(T_1, G_1)} a_1 \xrightarrow{(T_2, G_2)} a_2 \longrightarrow \dots$$

Narrative continuity. A higher constructor collects admissible unfoldings into coherent traces.

Attestation. Attestation is a dependent type over histories:

$$\text{Attest}(a) := \Sigma(h : \text{Hist}(a)). \text{Verify}(h),$$

classifying verifiable histories of an agent.

None of these require primitive declaration; they arise from the fixed point itself.

5 Properties

Three immediate properties follow:

1. **Minimality.** The formulation relies solely on the fixed-point equation (1).
2. **Universality.** Any agent architecture becomes an instance once its semantic operational law is specified.
3. **Pragmatic adequacy.** Graft embeds natural-language pragmatics directly into the semantic process.

6 Conclusion

By treating an agent as the fixed point of a semantic operational law acting on task structure and graft, agency is recast as generated rather than presupposed. Identity, lifecycle, narrative structure, and attestation arise without additional primitives. The formulation offers a minimal foundation capable of unifying linguistic pragmatics and computational perspectives on agency while remaining independent of implementation detail.