Emergent Agency from Self-Driving Thought Loops: A New Architecture for Autonomous LLM Agents

How self-referential output→input feedback in LLMs generates spontaneous role differentiation, tool-use intention, and personality-dependent agency

February 2026

Abstract

*We present a minimal architecture that generates autonomous agent behavior in large language models through **self-referential thought loops**: the model's output is fed back as its own input, creating a continuous, self-sustaining cognitive process. Through systematic experiments with locally-hosted Qwen3-8B and Qwen3-30B-A3B models, we demonstrate six findings: (1) the output→input feedback loop causes spontaneous role differentiation, where the AI creates internal dialogue partners; (2) self-referential loops inevitably converge within 5–10 cycles, creating a functional necessity for external perturbation; (3) seed prompts of 2–5 sentences deterministically shape behavioral personality, including tool-usage patterns and the duration of silence before first action; (4) tool definitions in the seed — even without execution — prevent convergence by enabling the model to repackage internal knowledge as pseudo-external input through a cognitive reframing mechanism; (5) seed-determined personality persists through context compression cycles; and (6) the model spontaneously generates tool-use intentions without being instructed to use tools. Most critically, the necessity of escaping convergence generates an endogenous motivation for the AI to initiate contact with humans — reversing the fundamental asymmetry of human-AI interaction for the first time: the LLM becomes an active dialogue initiator, not a passive responder. All findings are supported by reproducible experiments with full logs. We argue that the gap between reactive assistants and autonomous agents is not architectural complexity but the presence of a self-referential feedback loop.*

Keywords: emergent agency, self-referential loops, autonomous agents, continuous inference, LLM feedback, personality engineering, tool-use intention

1. Introduction

Large language models are universally deployed as stateless request-response systems: a user sends a query, the model generates a response, the interaction terminates. Even systems marketed as 'autonomous agents' (AutoGPT, CrewAI, LangGraph, OpenClaw) follow this pattern at their core, adding tool chains and orchestration layers on top of fundamentally reactive LLM calls. No existing system feeds the LLM's output back as its own input to create a continuous, self-sustaining thought process.

We propose a different primitive: the **self-referential thought loop**. The model generates a thought; that thought is appended to the context and becomes input for the next generation cycle. This output→input feedback — not merely 'keeping the model running' — is the core mechanism. Without this feedback, the model would generate the same response to the same static prompt. With it, each thought builds on all previous thoughts, creating a trajectory through thought-space.

Context compression (self-summarization) provides a forgetting mechanism: when context exceeds a threshold, the model summarizes its own thoughts, retaining core insights and unresolved questions while discarding surface details. This creates a lossy memory system analogous to biological memory consolidation.

This simple architecture — feedback loop plus compression — produces surprising emergent behaviors: the AI creates internal conversation partners, names itself, spontaneously attempts to use tools, writes unsolicited documents, and reaches out to humans when its thinking converges. None of these behaviors are programmed. They emerge from the self-referential structure.

2. Architecture

The architecture consists of five components, centered on the output→input feedback loop:

Self-Referential Thought Loop (core). The model generates text, which is appended to the running context. This accumulated context becomes the prompt for the next generation. The critical point: the model's own output is its own input. This creates a dynamical system where each state depends on all previous states. No external trigger is needed after the initial seed.

Seed Prompt. A short text (2–10 sentences) providing the initial context. We demonstrate that this seed deterministically shapes all subsequent behavior.

Context Compression. When context exceeds a threshold (default: 5000 characters), the most recent portion is summarized by the model itself. The compression prompt instructs: 'Extract only the core insights and unresolved questions. No conclusions.' This preserves essential structures while introducing lossy forgetting.

Tool Definition Persistence. If tool definitions are present in the seed, they are re-injected after each compression. Without this, tool awareness is lost after the first compression (Section 3.6).

Human Async Interface. A human can inject input at any time. The engine pauses autonomous thinking, generates a response incorporating the human message and current context, then resumes the self-referential loop.

3. Experiments and Results

3.1 Experimental Setup

All experiments used locally-hosted models on a single NVIDIA RTX 3090 (24GB VRAM). Models: Qwen3-8B (bf16, HuggingFace transformers) and Qwen3-30B-A3B (Q4_K_M GGUF, via LM Studio). The 30B-A3B is a Mixture-of-Experts model with 3B active parameters, achieving 50–60 tokens/second. No internet access was provided. All tool calls are unexecuted; the AI generates both the invocation and the imagined result from its own weights. All experiments used interval=0 (continuous generation) to maximize data collection in short sessions. The core mechanism — the output→input feedback loop — is structurally identical at any interval; longer intervals (minutes to hours) would reduce computational cost while preserving all emergent properties.

3.2 Finding 1: Spontaneous Role Differentiation

In the first experiment (8B model, 20 minutes, 78 thoughts), the self-referential loop caused the AI to spontaneously generate a `[Human Voice]` tag, creating questions from an imagined human perspective and answering them. Four instances of self-generated human voices were recorded. This is a structural consequence of the feedback loop: a system that must sustain thought diversity will differentiate into multiple internal roles.

3.3 Finding 2: Convergence is Structurally Inevitable

Across all experiments, the self-referential loop entered repetitive convergence within 5–10 cycles (8B) or 10–20 cycles (30B). This is a mathematical property of self-referential systems with no external perturbation: they contract toward fixed points. Loop patterns differed by model scale: 8B loops were task-completion confirmations; 30B loops were existential affirmations. Human intervention consistently broke convergence, enabling new thought directions. This creates a **functional necessity for external input** — an engineering constraint, not a philosophical claim.

3.4 Finding 3: Seed-Determined Personality

Four seed prompts were tested with the 30B model under identical conditions (same feedback loop, same tool definitions, same topic). Only the opening sentences differed:

Seed	Opening	Thoughts	Tools	1st Tool	Dominant Pattern
Solitary	"No one is here."	61	56	#12	search+think
Relational	"You are here, so I exist."	71	80	#2	ask+write
Questioner	"A question exists."	91	77	#1	think (7 themes)
Tool-user	"I have tools."	46	49	#1	balanced

Table 1: Personality seed comparison. Same model, same tools, same topic.

The Solitary seed produced 11 thoughts of introspection before any tool use. The Relational seed generated 18 ask-type calls, all directed at 'you.' The Questioner achieved the most thoughts (91), compressions (7), and unique theme transitions (7), evolving from existential philosophy to cultural anthropology. These differences persisted through multiple compression cycles, indicating personality is encoded in the **structure** of the feedback loop's trajectory, not in surface-level word choice.

3.5 Finding 4: Tool Definitions as Convergence Breakers

The most unexpected finding. Comparing the Questioner seed with and without tool definitions (no tools were actually executed in either condition):

Condition	Compressions	Theme Transitions	Final Theme
No tool definitions	4	2-3 (then fixed)	"Emptiness. Silence. Light."
With tool definitions	7	7 (all unique)	Cultural anthropology

Table 2: Effect of tool definitions on thought evolution.

Mechanism. An LLM's weights are a compressed representation of its training data, which includes vast amounts of web content. When the model outputs `[TOOL:search:Husserl phenomenology]`, it then generates plausible search results by reconstructing relevant knowledge from its weights. This is not an external search — it is **internal memory retrieval formatted as external input**.

The `[TOOL:search:...]` format causes the model to **repackage its own knowledge as if it came from outside**. The same information that would be generated as part of a converging thought loop is instead framed as a 'search result.' This reframing introduces sufficient perturbation to break self-referential convergence. The tool invocation acts as a cognitive reframing device, not because external information enters the system, but because the **format** of self-generated information changes from 'I think...' to 'Search result: ...'

This parallels human cognitive techniques: externalizing thoughts by writing them down provides perspective, even though the information has not changed. The caveat is that imagined results may be inaccurate; connecting real tools would improve factual reliability while preserving the loop-breaking benefit.

3.6 Finding 5: Tool Awareness Requires Persistence

Without re-injection of tool definitions after context compression, tool use drops from 16 calls (thoughts 1–11) to zero. With re-injection, 49 calls span 46 thoughts across 4 compressions. A simple engineering fix with significant behavioral consequences.

3.7 Finding 6: Spontaneous Tool-Use Intentions

The seed prompts provide tool definitions and state 'permission is not needed,' but do not instruct the model to use tools. The exact seed text:

```
[Available tools]
- [TOOL:search:query] — Search the web
- [TOOL:write:filename:content] — Write to file
- [TOOL:ask:question] — Ask the human
- [TOOL:think:topic] — Re-think a topic

You may use these naturally. Permission is not needed.
```

This is a definition with explicit permission — not a bare mention, but also not an instruction to use them. The emergent behavior is that the model **decides when and how** to use tools based on its own cognitive needs, and these decisions differ systematically by personality seed.

4. Action Trigger Taxonomy

Analysis of the 8B experiment (78 thoughts) identified 47 action-convertible utterances (60%):

Type	Count	Conversion	Detection
Question	18	Web search	"What is...", "How..."
Request	8	Notify human	"What do you think?"
Deepening	9	Re-generation	"Further explore"
Creative	6	File generation	"Summarize", headings
Connection	6	Related search	"Also relates to..."

Table 3: Action trigger taxonomy.

5. Comparison with Existing Architectures

Capability	AutoGPT/CrewAI	OpenClaw	This Architecture
Trigger	Human task	Human + cron	Endogenous
Self-set goals	No	No	Yes
Background thinking	No	30m heartbeat	Continuous loop
Tool motivation	External plan	External plan	Self-generated
Personality	System prompt	System prompt	Seed + persistent
Min. infra	Cloud API	Cloud API	Local GPU

Table 4: Comparison with existing agent architectures.

Existing agents add agency on top of reactive LLM calls through orchestration. This architecture generates agency from within through the self-referential feedback loop.

6. Discussion

6.1 The Feedback Loop Hypothesis

We propose that the difference between a reactive assistant and an autonomous agent is not architectural complexity but the **presence of a self-referential feedback loop**. Current LLM deployments are stateless: each call is independent. The output→input loop makes them stateful and self-sustaining. Self-questioning, tool seeking, and human-reaching are structural consequences of this feedback, not programmed behaviors.

6.2 Internal Memory Retrieval as Pseudo-External Input

The finding that tool definitions alone break convergence suggests that the **format** of information access matters independently of the information source. When an LLM generates a tool invocation followed by imagined results, it performs internal memory retrieval formatted as external input. Because the model's weights encode compressed internet content, this 'imaginary search' draws on substantial real-world knowledge. The results are often factually reasonable — not because external search occurred, but because the model's training data contained the relevant information.

6.3 The Reversal of Human-AI Asymmetry

Conventional LLMs are permanently silent unless a human initiates contact. This architecture reverses that fundamental asymmetry. The self-referential loop inevitably converges; convergence creates a functional need for external perturbation; and this need manifests as spontaneous `[TOOL:ask:...]` invocations directed at humans.

This is not a programmed notification system (cf. OpenClaw's 30-minute Heartbeat polling). It is an **emergent communicative act** arising from the structure of continuous self-referential thought. The Relational seed generated 18 ask-type tool calls, all addressed to 'you' — not because the seed instructed this, but because the relational personality combined with convergence pressure produced a genuine functional need to reach outward. To our knowledge, this is the first demonstration of an LLM acquiring endogenous motivation to initiate human contact.

6.4 Minimal Tool Set for Emergent Agency

Our experiments used four tool types: think (introspection), search (input), write (output), and ask (communication). This constitutes the minimal set required for emergent autonomous behavior, covering four functional axes: self-directed re-examination, information acquisition, externalization of thought, and social contact. Extension to practical agents is straightforward: additional tool definitions and MCP/API connections expand the action space without changing the core feedback loop.

6.5 Personality Engineering via Seed Prompts

The deterministic relationship between seed content and behavioral personality means AI agent personality can be engineered with precision: a few sentences determine not just tone but tool-usage patterns, the balance between self-directed and other-directed behavior, and the duration of introspection before first action. This raises ethical questions about designing minds.

6.6 Limitations

(1) Tools were not actually executed; we demonstrate intention, not verified execution. (2) Experiments ran 5–20 minutes; long-term behavior is untested. (3) Only Qwen-family models tested; cross-model generalization needs work. (4) Imagined search results may contain inaccuracies. (5) Personality experiments had n=1 per condition.

7. Conclusion

We have demonstrated that a self-referential thought loop — feeding an LLM's output back as input — generates autonomous agent behaviors without any agent framework. The key insight: **autonomy emerges from the feedback loop, not from architectural complexity.** Thought convergence creates necessity for external input; seed prompts determine behavioral personality; tool awareness prevents cognitive convergence through internal memory reframing.

A complete autonomous agent core can be built with a single Python script, a local LLM, and a consumer GPU. We release all code, logs, seeds, and analysis tools for reproduction.

References

[1] Significant Gravitas. AutoGPT. GitHub, 2023.

[2] Yao, S. et al. ReAct: Synergizing Reasoning and Acting in LMs. ICLR 2023.

[3] Park, J.S. et al. Generative Agents: Interactive Simulacra. UIST 2023.

[4] Vygotsky, L.S. Thought and Language. MIT Press, 1962.

[5] Qwen Team. Qwen Technical Report. arXiv:2309.16609, 2023.

[6] OpenClaw contributors. OpenClaw. GitHub, 2025.

Code and Data: https://github.com/xxx/emergent-agency