

CONTEXT PROPAGATION SYSTEM

Causal continuity infrastructure for artificial worlds

Events become evidence. Evidence becomes actor-local meaning. Meaning can become action. Accepted consequences remain available to shape what happens next.

WOJCIECH HUPERT

VERSION 1.0 / AUGUST 2026

CONTEXTPROPAGATIONSYSTEM.COM

Contents

Contents	1
Abstract	2
1. The continuity problem	2
2. The CPS thesis	3
3. A closed causal loop	3
4. Six pillars, one system	4
5. The cross-pillar action spine	5
6. Memory, belief, and propagation	5
7. Reflection and later behavioral effect	6
8. Social interaction as a governed lifecycle	6
9. Ordinary life and actor-created artifacts	7
10. Providers without semantic authority	7
11. CPS Studio and causal observability	8
12. Persistence, replay, and lifecycle	8
13. Lelit Distrikt and embodied clients	9
14. Evaluation model	9
15. Application areas	10
16. Relationship to adjacent approaches	10
17. Limits and non-claims	10
18. Conclusion	11
References	11

Abstract

The Context Propagation System (CPS) is an operational framework for building persistent, inspectable artificial worlds in which information has social and causal consequences. It connects a shared World, actor-local memory and interpretation, social interaction, action arbitration, external model providers, and human-facing observability through one durable causal loop.

CPS addresses a problem that fluent dialogue alone does not solve. A believable artificial character must not only produce an appropriate line. The surrounding system must know what happened, who could perceive it, what each actor made of it, whether any belief, concern, goal, relationship, or memory changed, what action became eligible, whether that action was admitted and executed, and which later consequence can be attributed to it. Those distinctions must survive pauses, retries, restarts, provider failures, and long-running simulation time.

The system therefore treats language models as bounded realization and interpretation providers, not as authorities over world truth. Canonical state belongs to explicit domains. Objective location belongs to World. Subjective memory and meaning belong to Anima. Interaction legitimacy belongs to Forum. Work selection belongs to Scheduler. Read models belong to Observatory and cannot mutate the simulation. Core coordinates and records accepted transitions across those boundaries.

The current implementation demonstrates recurring ordinary life, movement, durable Player-to-Actor and Actor-to-Actor communication, actor-local Memory, bounded Reflection with later behavioral effect, relationship-sensitive social continuation, source-grounded work artifacts, historical inspection, recovery-safe external attempts, and CPS-side spatial and Player integration for an embodied client. Lelit Distrikt is the first authored world built on this architecture; CPS Studio is its control, participation, and explanation surface.

This paper describes CPS as a working bounded system. "Bounded" is a design property: the world operates through authored identities, validated data, explicit policies, typed state transitions, and inspectable evidence. CPS does not claim consciousness, unrestricted autonomy, or a universal model of society.

1. The continuity problem

Generative dialogue makes individual moments easy to produce. Continuity makes those moments difficult to govern.

In a simple chat system, a user message enters a model and a response returns. Context may be retained as text, embedded for retrieval, or summarized for a later prompt. This can create conversational coherence, but it does not by itself create a world. A world requires separate answers to questions that a transcript tends to collapse:

- Did the event actually occur, or was it only proposed?
- Which actors had access to it, and which did not?
- What did each actor remember, believe, doubt, or misunderstand?
- Was information private, relationship-bound, or socially mobile?
- Why did an actor speak, wait, move, continue, or stop?
- Which candidate action won when several demands competed?
- Did a provider attempt fail, or did the world reject its result?
- Which accepted consequence can influence a later decision?

Without explicit answers, apparent social behavior is easily produced but hard to trust. A model can narrate that a character moved, learned, promised, completed work, or changed a relationship without any of those changes becoming valid system state. It can also produce plausible explanations after the fact that do not correspond to the

actual decision path.

CPS makes continuity a first-class runtime problem. It replaces the idea of one ever-growing prompt with a governed chain of evidence, actor-local state, eligible action, canonical consequence, and provenance.

2. The CPS thesis

The central thesis of CPS is simple:

Conversation is evidence of a social system, not the social system itself.

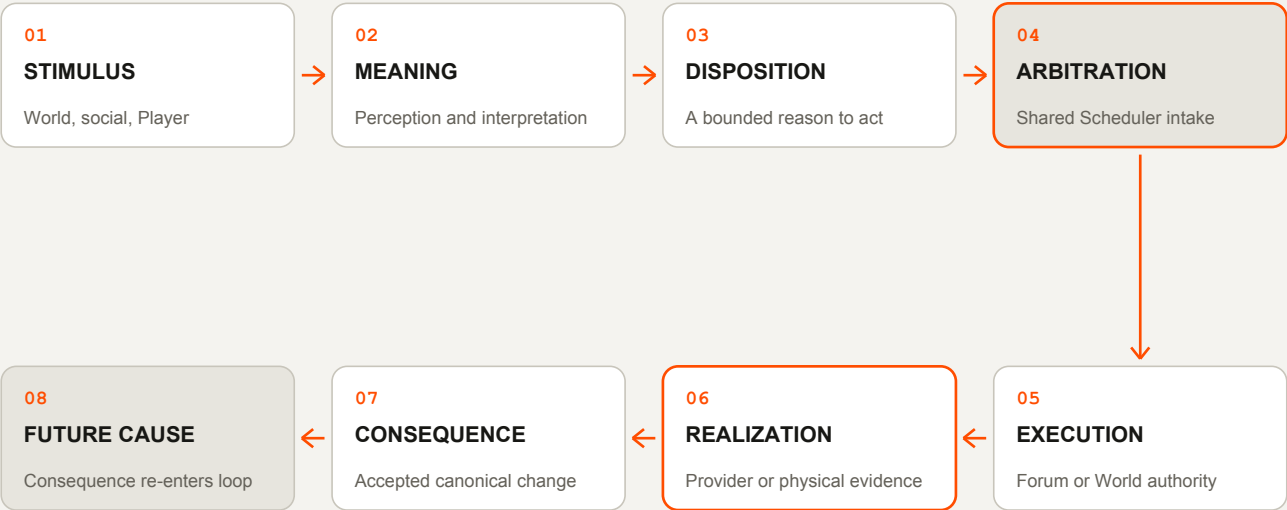
Dialogue is one visible output of a larger process. The durable object is the changing relationship between shared reality and actor-local reality over time. CPS therefore models social state rather than optimizing only for isolated conversational quality.

This produces five design commitments.

- 1
- Truth has an owner. Shared facts, actor beliefs, interaction state, work state, and UI summaries are not interchangeable.
- 2
- Consequences require acceptance. A generated claim or device acknowledgement is evidence until the appropriate authority validates it.
- 3
- Actors are locally situated. Each actor receives partial evidence, forms local meaning, and may remain uncertain or unaware.
- 4
- Action is arbitrated. A reason to act does not bypass current work, conflicts, availability, or interaction rules.
- 5
- Every meaningful transition is inspectable. A person can move from a readable explanation to the exact evidence and lineage behind it.

These commitments distinguish CPS from a chatbot with memory, a vector database, a generic game engine, or an unstructured multi-agent prompt loop. CPS can use all of those technologies at its edges, but none of them substitutes for its authority model.

3. A closed causal loop



CPS is organized around one recurring loop:

```
stimulus or accepted consequence
-> access and attention
-> actor-local perception and interpretation
-> state reassessment
-> disposition or work demand
-> shared runnable intake
-> Scheduler arbitration
-> Forum or World execution
-> provider realization or physical evidence
-> accepted canonical consequence
-> future stimulus
```

The final transition matters most. A system that reacts to an event and then stops is an event-response engine. A persistent world must make the accepted result available to later processing.

Consider a bounded example from Lelit Distrikt. An actor completes a routine-driven movement to a cafe. World accepts the arrival and updates objective location. Another actor has access to the co-presence event and notices the arrival. Anima interprets the person who arrived and evaluates whether an exact retained relationship consequence makes that presence significant. If it does, the actor may form a disposition to communicate. The candidate enters the same Scheduler used by other work. Forum verifies that the interaction is admissible. A text provider realizes an already-selected social action. The accepted turn becomes evidence for the recipient, who may interpret and retain it. The resulting state can later matter again.

Every arrow names a distinct status and a distinct authority. The system can therefore say where the chain stopped without fabricating a motive or result.

4. Six pillars, one system

CPS separates conceptual ownership into six pillars. These pillars are not necessarily services or code packages. They are stable answers to the question: who owns the meaning of this state?

Pillar	Responsibility	Examples of owned truth
Core	Coordinate, validate, persist, and record transitions	events, snapshots, evidence, canonical records, provenance, runtime outcomes
World	Represent shared external reality	locations, objects, institutions, authored conditions, environmental pressure, accepted movement
Anima	Represent actor-local subjective reality	perception, Memory, beliefs, uncertainty, appraisal, goals, values, actor-local relationships
Forum	Govern social interaction	participants, channels, admission, turn-taking, purpose, continuation, interruption, closure
Soma	Represent embodied condition and capability	bodily state, sensory capability, posture, gesture, physical action
Observatory	Make state and causality understandable	read models, timelines, explanations, state differences, source trails, monitoring

Core is deliberately neutral. It can carry a World transition without owning spatial meaning, or persist an Anima state version without deciding what an actor believes. World is the sole authority for objective location. Anima may hold an actor's mistaken belief about location, but that belief does not relocate anyone. Forum can admit a conversation without inventing the actor's reason for speaking. Observatory can display the entire chain without gaining permission to

change it.

Soma is the embodiment boundary. It is designed to own bodily capability and condition while World retains objective spatial truth. This prevents an engine, robot, animation system, or device from becoming the semantic authority merely because it executed or displayed an action.

5. The cross-pillar action spine

The pillars are connected by a shared action spine. This is where CPS turns a reason or demand into an executed consequence.

5.1 Actor-local meaning

Anima records what an actor could perceive, how the actor interpreted it, and whether local state changed. Its outputs may include a Memory, uncertainty, an appraisal, a Goal, a relationship-relevant evaluation, or a disposition to act. A disposition records a bounded answer to "what is this actor now disposed to do, and why?" It is not a command and not a general plan.

5.2 Runnable work and arbitration

Immediate actions and durable Tasks enter a shared runnable vocabulary. Scheduler owns selection, blocking, deferral, and competition. This ensures that routine movement, social communication, Player-request follow-through, and other supported work do not create independent selection authorities.

5.3 Domain execution

After selection, the correct domain retains execution authority. Forum admits and structures communication. World authorizes and applies movement or shared-state consequences. A provider may realize selected text or return physical execution evidence, but cannot promote its own output into canonical state.

5.4 Consequence and recurrence

Accepted outcomes are persisted with source identity, simulation time, policy identity, and causal lineage. They can be projected to people and re-enter the world as future stimuli. This closes the loop without an alternate activation queue or an opaque agent controller.

6. Memory, belief, and propagation

CPS treats memory as actor-owned evidence, not as globally retrieved text. A Memory version can name its owner, source, claim, privacy or propagation constraints, confidence, time, and lineage. Different actors can hold different records about the same event. Dialogue received from another actor is evidence to evaluate, not truth to copy.

The intended information path separates several questions:

- 1 What evidence entered the system?
- 2 What claim or social content can be extracted from it?
- 3 Which actor receives a Memory, if any?
- 4 Is the material private, protected, actor-local, public, relationship-relevant, or otherwise constrained?
- 5 Does it conflict with existing belief or uncertainty?

- 6 Is it salient enough to reconsider?
- 7 Is there a reason to communicate it to this recipient now?
- 8 Does Forum admit the interaction?
- 9 What does the recipient retain or change after hearing it?

This separation prevents "retrievable" from meaning "shareable" and "mentioned" from meaning "believed." It also makes privacy a structural property rather than a prompt instruction.

7. Reflection and later behavioral effect

Reflection in CPS is bounded actor-local reassessment. It is not a transcript of hidden reasoning and not a background stream of unrestricted thought.

The implemented AR-1 path begins with an accepted direct-Player Memory. A typed Reflection occasion becomes eligible, enters normal arbitration, and may use a provider to propose a source-grounded reassessment. Anima validates that proposal and may adopt zero or one epistemic Goal. The accepted goal can alter a later social selection.

The important proof is counterfactual. In a source-present run, the Memory and Reflection lineage lead the actor to seek an answer to a Reflection question at a later Forum opportunity. In a source-absent control with the same authored opportunity, the actor selects the authored claim purpose instead. The difference is carried through direct source lineage rather than inferred from adjacent timestamps.

This gives Reflection a precise product meaning: retained experience can alter what later matters. It does not imply general planning, consciousness, or continuous inner monologue.

8. Social interaction as a governed lifecycle

CPS supports two coherent conversation modes.

Actor-to-Actor interaction is normally read-only from the Player's perspective. The system records participants, encounter identity, purpose, admission, selected semantic action, provider attempt, accepted turn, recipient interpretation, continuation, interruption, closure, and selected consequences.

Player-to-Actor interaction behaves like familiar messaging. Player-authored text is retained distinctly from provider output. The Actor may respond automatically when the runtime can continue. Sending, waiting, retry, failure, and unavailable states remain visible. A Player message can become actor-local evidence, but it does not automatically create a Memory, Goal, Task, relationship change, or World fact.

Forum owns the social lifecycle. It answers whether a particular interaction is legitimate and how it proceeds. The actor's reason comes from actor-local state; the timing comes from Scheduler; the wording comes from a provider after semantic selection; the accepted social consequence is committed by the appropriate authority.

Interruptions are similarly typed. A supported encounter may pause a visible routine occurrence and later resume it without rewriting the authored schedule. This allows social life to affect activity while preserving the difference between authored time and lived execution.

9. Ordinary life and actor-created artifacts

Artificial worlds become legible when actors do more than wait for conversation. CPS supports recurring authored routines that generate bounded ordinary-life demands. A routine interval can require an actor to be at a location. If objective World state differs, CPS derives Task-owned movement work, sends it through shared arbitration, and lets World apply the accepted arrival. The routine is a reason for work, not a teleport command and not a substitute for current location.

Recurring weekly routines allow the world to progress across multiple simulated days without rewriting authored definitions. Concrete occurrences receive distinct identities, while replay of the same boundary remains idempotent.

CPS also supports actor-created work artifacts. An actor's lived source material can produce an attributed working version such as a business note, journal fragment, or broadcast transcript. Each version retains an ordered source bundle, frontier, predecessor, revision reason, provider history, and acceptance state. A new simulated frontier without new source material records a truthful no-change result; it does not generate novelty for its own sake.

Artifacts therefore show continuity rather than only generation capability:

```

encounter or experience
-> retained source material
-> working version
-> later material
-> explicit revision evaluation
-> revised or unchanged artifact history

```

Provider failure, rejection, or restart ambiguity remains visible and cannot fabricate completed work.

10. Providers without semantic authority

Large language models are valuable in CPS because they can interpret bounded input, propose actor-local content, and realize natural language. They are also non-deterministic external systems that may be unavailable, malformed, stale, or semantically overreaching.

CPS therefore treats every provider call as an attempt with identity, status, input lineage, output, validation, and recovery semantics. A provider may:

- realize an already-selected communicative action;
- extract or consolidate bounded Memory content;
- propose a source-grounded Reflection;
- generate or revise an attributed artifact;
- summarize evidence for a projection.

A provider may not:

- decide that a World event occurred;
- invent an actor's canonical location;
- declare a Task complete;
- create a relationship consequence by prose alone;
- bypass Forum admission or Scheduler arbitration;
- silently repair malformed authored data;

- turn uncertain testimony into shared truth.

This makes CPS provider-portable. Quality, latency, cost, transport, timeout, and repair can be compared by operation family while the semantic contracts remain stable. Routing changes require explicit approved thresholds; the system does not infer a policy from one benchmark result.

11. CPS Studio and causal observability

CPS Studio is the human interface to the running world. It combines observation, participation, explanation, setup, temporal control, and technical inspection without making the interface authoritative over the state it displays.

Its primary product surfaces are:

- World, for current and historical shared state, activity, movement, and world vitals;
- Actor and Messenger, for actor-local inspection and Player participation;
- Communicator, for readable Player-to-Actor chat and Actor-to-Actor encounter inspection;
- Events and Timeline, for causal changes and movement through simulation history;
- System, for explanations, evidence coverage, alternatives, integrity, and source trails;
- Setup, for choosing and preparing an authored world and workspace;
- Life and Tasks, for bounded actor work, artifacts, and delegated work projections.

The interface follows progressive disclosure:

```
readable event or conversation
-> concise "why this?" explanation
-> state and decision summary
-> provenance and source trail
-> raw technical record on request
```

This order matters. A normal user should first understand who spoke, what happened, whether the exchange is active or failed, and what can happen next. An investigator should be able to continue down to encounter identity, semantic action, provider attempt, Memory source, policy version, causal edge, and raw record.

Observatory projections make missing evidence visible. They do not infer causality from timestamps, display names, or proximity. "Not recorded," "blocked," "ambiguous," and "unavailable" are valid outcomes, not UI gaps to fill with plausible text.

12. Persistence, replay, and lifecycle

Continuity is credible only when it survives operational seams. CPS isolates authored presets, editable workspaces, and simulation runs. A protected preset package provides a validated baseline. A workspace receives its own persistent state. Runs carry their own temporal identity, snapshots, ledgers, checkpoints, and provider attempts.

Every causal stage is designed around stable identity, simulation time, source lineage, idempotent re-entry, and explicit recovery state. A restart must not duplicate a conversation, movement, Task completion, Memory consequence, or artifact version. Ambiguous external dispatch is not silently replayed. Stale callbacks cannot mutate a newer processing attempt.

The authored-data boundary is strict. Actor IDs, location IDs, routine fields, and other authored values are preserved exactly. Missing or invalid authored data fails validation before snapshot creation. The runtime does not normalize,

coerce, enrich, or invent a fallback meaning in order to keep the simulation moving.

Time is similarly explicit. CPS distinguishes wall time, simulation time, pause state, current frontier, and catch-up or recovery behavior. New product runs begin paused so an operator can inspect and deliberately resume the world. Recurrence derives concrete intervals from immutable authored schedules rather than rewriting the source canon.

13. Lelit Distrikt and embodied clients

Lelit Distrikt is the first operational world for CPS. It supplies a bounded Berlin district, authored residents, identities, routines, relationships, locations, and artifact capabilities. Its purpose is not to make CPS specific to one story. It provides enough social and temporal structure for causal continuity to become visible.

In Lelit, ordinary life can recur, actors can move between authored locations, notice one another, communicate, retain direct evidence, reflect on selected experience, pursue bounded epistemic goals, revise personal work, and expose the causal trail in Studio. Differences between actors come from authored data and retained state, not Actor-ID branches in runtime logic.

CPS also defines a clean bridge to an embodied client such as Unreal Engine. Unreal can author and export a versioned spatial revision, submit Player presence against the exact run-bound revision, render actors and environments, handle input, and realize presentation. CPS retains semantic authority for actor identity, current accepted location, conversation, Memory, work, and social consequence. Voice wraps canonical accepted text; it does not create a parallel dialogue path.

This division allows a visual world to be rich without making its renderer the owner of social truth.

14. Evaluation model

CPS is evaluated as a causal system, not only by whether generated dialogue sounds plausible. A meaningful acceptance test combines positive behavior, negative behavior, counterfactual control, recovery, and inspectability.

Representative evaluation questions include:

- Does an accepted source produce a later behavioral difference when the source-absent control does not?
- Can two kinds of work compete through one Scheduler without duplicate execution?
- Does Forum rejection prevent a provider call and canonical turn?
- Does a provider failure remain a failure rather than a successful-looking transcript?
- Does a relationship-present case differ from a relationship-absent case without introducing a generic trust threshold?
- Does repeated artifact evaluation with the same source bundle produce no new version?
- Can a restart occur between decision and execution without duplicating consequences?
- Can a person trace the result through exact direct lineage rather than inferred chronology?
- Do invalid authored identifiers fail before any snapshot exists?

The current reference implementation has been exercised through focused causal suites, full backend regression, frontend interaction checks, production builds, checkpoint and migration tests, provider failure tests, and bounded multi-actor scenarios. The purpose of this evidence is not to claim universal social realism. It is to show that the stated contracts operate together and that failure boundaries remain truthful.

15. Application areas

CPS is useful wherever artificial characters or agents must share a persistent world without collapsing all state into a model prompt.

Interactive and embodied worlds

Games, simulations, XR experiences, robots, and installations can use CPS to connect conversation, location, relationship, memory, and later consequence while retaining an external engine for rendering and physical execution.

Narrative and cultural installations

Visitors can enter a bounded social environment where an exchange affects what particular actors later remember, disclose, withhold, create, or misunderstand. The operator can inspect how the world reached that result.

Agent reliability research

CPS offers a controlled environment for studying memory attribution, propagation, privacy containment, belief divergence, provider failure, recovery, and causal auditability. The strict separation between model output and accepted state makes failure analysis concrete.

Training and rehearsal environments

Authored communities can preserve roles, institutional conditions, prior encounters, and participant actions across sessions. A later response can be traced to retained evidence rather than justified only by model fluency.

Creative production systems

Actor-created artifacts can accumulate source-grounded revisions over simulated time. This supports characters whose work reflects their lived history instead of appearing as isolated generations.

16. Relationship to adjacent approaches

CPS shares concerns with generative-agent architectures that combine memory, reflection, planning, and action to produce believable behavior [10]. Its emphasis is different. CPS treats authority, admission, arbitration, canonical consequence, failure semantics, and human inspectability as primary runtime contracts. Memory retrieval is one input to the loop, not the loop itself.

CPS also aligns with the broader provenance principle that data should retain information about the entities, activities, and agents involved in producing it so its reliability can be assessed [11]. CPS applies that principle to social simulation: a belief, Memory, utterance, artifact, movement, and explanation retain distinct source and acceptance histories.

The result is best understood as causal continuity infrastructure. A language model supplies expressive and interpretive capability. CPS supplies the durable world around it.

17. Limits and non-claims

CPS is intentionally not:

- a claim of consciousness, sentience, or human-equivalent understanding;

- a universal theory or faithful simulation of real society;
- an unrestricted planner or open-ended autonomous-agent framework;
- a guarantee that generated dialogue is compelling or socially correct;
- a replacement for a rendering engine, physics engine, robot controller, or content-authoring workflow;
- a system in which every event must create Memory, belief change, conversation, or action.

Its strength comes from explicit scope. Silence can be correct. No-change can be correct. A blocked action can be correct. Uncertainty can remain unresolved. A provider can fail without the world pretending otherwise.

18. Conclusion

The Context Propagation System turns artificial-world continuity from an implicit prompt-management problem into an explicit causal architecture.

It gives each kind of truth an owner. It lets actors remain locally situated rather than globally informed. It routes reasons and demands through shared arbitration. It keeps external providers expressive but non-authoritative. It preserves the difference between planned, attempted, accepted, and canonical. It makes the resulting life readable in ordinary product language and auditable down to exact sources.

The practical outcome is a world in which information can persist, remain private, change perspective, motivate action, produce work, and return later as consequence. Lelit Distrikt demonstrates that model in a bounded living setting. CPS Studio makes the setting observable and participatory. Embodied clients can render it without owning its social meaning.

CPS is therefore not a memory feature attached to dialogue. It is the continuity layer that allows dialogue, action, and time to become one inspectable world.

References

- 1 CPS Core v2, [Current State Dashboard](#), updated 23 August 2026.
- 2 CPS Core v2, [Canonical Causal Loop Contract](#), 10 August 2026.
- 3 CPS Core v2, [Architectural Pillars](#), accepted conceptual ownership matrix.
- 4 CPS Core v2, [Definition and Boundary](#).
- 5 CPS Core v2, [September Joint Acceptance Criteria and Sprint Plan](#), 2026.
- 6 CPS Core v2, [Architecture Convergence Acceptance](#), 23 August 2026.
- 7 CPS Core v2, [Integrated First-Phase Observability Evidence](#), 23 August 2026.
- 8 Context Propagation System, [CPS at a glance](#), public project one-pager, 2026.
- 9 CPS Core v2, [Pillars, Subsystems, and Action Mechanisms for Product Designers](#), 2026.
- 10 J. S. Park et al., [Generative Agents: Interactive Simulacra of Human Behavior](#), Proceedings of UIST 2023.
- 11 W3C, [PROV-O: The PROV Ontology](#), W3C Recommendation, 30 April 2013.

Copyright 2026 Context Propagation System. Created by Wojciech Hupert.