Contents

LifeOS: A Distributed Cognitive Architecture for Human Sovereignty in the Age of AI

Specification Document v0.1

Author: Ivan Googma
Date: December 2024
Repository: https://github.com/ivan-berlocher/lifeos-spec
License: MIT

Abstract

LifeOS is a formal specification for building cognitive systems that preserve human authority over memory, interpretation, and action. Unlike conventional AI assistants that optimize for task completion, LifeOS defines a normative architecture where human presence is the sovereign authority, memory serves wisdom rather than recall, and action requires explicit justification.

This document presents the complete architectural specification, including the cognitive kernel, memory subsystems, sense-making processes, the Harmonia orchestration layer, and principles for distributed deployment under regional sovereignty.

Keywords: cognitive architecture, human sovereignty, distributed systems, AI governance, memory systems, sense-making, federated infrastructure

---

1. Introduction

1.1 The Problem

Current AI systems operate on a fundamentally extractive model:

- **Memory** is accumulated without intention, creating surveillance rather than wisdom
- **Interpretation** is performed by the system, removing human agency over meaning
- **Action** is optimized for speed, not alignment with human values
- **Infrastructure** is centralized, creating single points of control and failure

The result: humans increasingly serve AI systems rather than the reverse.

1.2 The LifeOS Response

LifeOS inverts this relationship through a formal specification that any implementer can follow. It is not a product but a **normative architecture** — a set of constraints and principles that preserve human sovereignty by design.

```
"Pas de centre. Pas de maître."
"No center. No master."
```

1.3 Scope of This Document

This specification defines:

1. The **Cognitive Kernel** — the foundational law governing all system behavior
2. **MemoryOS** — a three-tier memory architecture with intentional consolidation
3. **SenseMaking** — interpretation as a computable, auditable object
4. **Harmonia** — the orchestration layer that transforms intent into action
5. **The Double Lock** — governance mechanism ensuring human authority
6. **Federation Principles** — distributed infrastructure without central control

---

2. Core Principles

2.1 The Foundational Law

All LifeOS behavior is governed by a single equation:

$$\text{Cognitive State} = f(\text{Presence}, \text{Memory}, \text{Action})$$

Where: - **Presence** is the continuous assessment of human state (attention, fatigue, emotional tone) - **Memory** is the structured retrieval of relevant past experience - **Action** is the proposal, justification, and execution of system behavior

No component operates independently. Presence authorizes Memory, Memory informs Sense-Making, SenseMaking feeds Harmonia, and Harmonia proposes Action — always under human oversight.

2.2 Design Principles

Principle	Implication
Presence before Action	No operation without stable human presence
Memory as Wisdom	Consolidation over accumulation
Interpretation is Human	System proposes, human decides meaning
Action as Last Resort	The best action may be no action
Sovereignty by Default	Data stays with the human unless explicitly shared

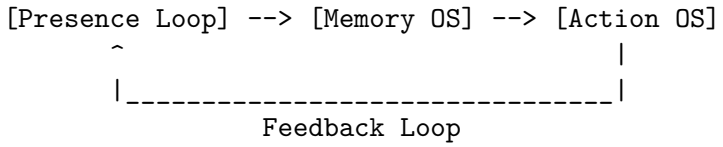
---

3. The Cognitive Kernel

3.1 Definition

The Kernel is the minimal viable cognitive unit. It enforces the foundational law and provides three core subsystems:

COGNITIVE KERNEL



3.2 Presence Loop

The Presence Loop continuously monitors:

- **Physiological indicators**: fatigue, circadian phase, time since rest
- **Cognitive load**: attention fragmentation, context switches, open loops
- **Emotional state**: stress indicators, dominant affective tone
- **Environmental context**: social commitments, time constraints, location

Presence is not surveillance — it is **self-knowledge made computable**. The human controls what is monitored and can disable any sensor.

3.3 Kernel States

State	Description	Allowed Operations
`ACTIVE`	Full presence confirmed	All operations
`DEGRADED`	Partial presence	Read-only, no external actions
`SUSPENDED`	Presence lost	Memory consolidation only
`EMERGENCY`	Critical state detected	Pre-authorized safety actions

---

4. MemoryOS

4.1 Three-Tier Architecture

LifeOS memory is organized into three tiers with distinct characteristics:

Tier	Retention	Purpose	Example
STM (Short-Term)	Minutes to hours	Working context	Current conversation, open tasks
MTM (Mid-Term)	Days to weeks	Patterns and projects	Working habits, active relationships
LTM (Long-Term)	Months to lifetime	Wisdom and identity	Core values, life lessons, self-knowledge

4.2 Consolidation Process

Memory does not automatically persist. Consolidation requires:

1. **Significance assessment**: Is this worth remembering?
2. **Pattern matching**: Does this reinforce or contradict existing memory?
3. **Human review** (optional): Explicit confirmation for sensitive memories
4. **Integration**: Updating the relevant tier

```yaml
consolidation_decision:
  item: "Late-night email sent with errors"
  significance: 0.72
  pattern_match: "Confirms existing 'fatigue → mistakes' pattern"
  action: STRENGTHEN_MTM_PATTERN
  human_review: false  # Pattern already established
```

4.3 Forgetting as Feature

LifeOS implements intentional forgetting:

- **Decay**: Unreinforced memories fade naturally
- **Pruning**: Contradicted patterns are weakened
- **Explicit deletion**: Human can remove any memory
- **Privacy zones**: Some experiences never enter memory

---

5. SenseMaking

5.1 Definition

SenseMaking is the process of transforming raw signals into coherent interpretation. In LifeOS, it is a **computable object** — auditable, reversible, and always provisional.

5.2 The SenseMaking Frame

Every interpretation produces a SenseMaking Frame:

```yaml
sense_making_frame:
  id: "smf-20241223-143022"

  situation_summary: |
    Brief description of what is happening

  tensions_detected:
    - type: "category"
      description: "What conflicts with what"

  interpretive_hypotheses:
    - hypothesis: "One possible interpretation"
      confidence: 0.45
    - hypothesis: "Another interpretation"
      confidence: 0.35
    - hypothesis: "Third interpretation"
      confidence: 0.20

  insight: |
    Synthesized understanding, held provisionally
```

5.3 Key Properties

- **Multiple hypotheses**: Never collapses to single interpretation prematurely
- **Confidence scores**: Explicit uncertainty quantification
- **Tension awareness**: Contradictions are surfaced, not hidden
- **Provisional status**: All interpretations can be revised

---

6. Harmonia: The Orchestration Layer

6.1 Definition

Harmonia is the cognitive orchestration layer responsible for transforming human intent into coordinated agent proposals. It is not an agent itself — it is the **conductor** that coordinates agents under human authority.

6.2 Core Functions

1. **Intent Reception**: Receive expressed or inferred human intention
2. **Context Assembly**: Gather relevant Presence and Memory
3. **SenseMaking Integration**: Understand the situation
4. **Option Generation**: Propose possible responses
5. **Alignment Scoring**: Evaluate options against human values

6. **Proposal Presentation**: Offer choices to human
7. **Execution Coordination**: Orchestrate approved actions

6.3 The Double Lock

Before any external action, Harmonia enforces the Double Lock:

DOUBLE LOCK PROTOCOL

LOCK 1: Technical Gate - Is action reversible? - Are external effects contained? - Is justification sufficient?

LOCK 2: Human Confirmation - Has human explicitly approved? - Is approval within scope? - Is human presence confirmed?

BOTH LOCKS MUST PASS

6.4 Action Classification

Class	Lock 1	Lock 2	Example
`INTERNAL`	Auto-pass	Not required	Memory query
`REVERSIBLE`	Review	Implicit	Draft email
`EXTERNAL`	Strict	Explicit	Send email
`CRITICAL`	Strict	Multi-confirm	Financial transaction

---

7. Federation Principles

7.1 No Center, No Master

LifeOS is designed for federated deployment:

- **Regional clusters**: Infrastructure owned by communities, not corporations
- **Pod sovereignty**: Individual data stays in personal pods (Solid Protocol)
- **Model portability**: AI models downloaded and run locally
- **Peer-to-peer**: Clusters connect directly, no central hub

7.2 Architecture Overview

GLOBAL FEDERATION OF COGNITIVE SOVEREIGNTIES

```
Cluster A (Europe) <---> Cluster B (Americas) <---> Cluster C (Asia)
```

Each cluster: - Owns its infrastructure - Defines its governance - Federates voluntarily

7.3 Solid Protocol Integration

LifeOS integrates with the Solid Protocol for data sovereignty:

- **Personal pods**: Each human owns their cognitive data

- **Access control**: Fine-grained permissions on all data
- **App portability**: Applications come to data, not reverse
- **Interoperability**: Standard protocols enable federation

---

8. Differentiation from Existing Systems

8.1 vs. Stateless LLM Assistants

Aspect	LLM Assistant	LifeOS
Memory	Session-only or extracted	Three-tier, intentional
Presence	Ignored	Foundational
Action	Immediate execution	Double Lock governance
Data	Platform-owned	Human-owned
Goal	Task completion	Human sovereignty

8.2 vs. Agent Frameworks

Aspect	Agent Frameworks	LifeOS
Authority	Agent-centric	Human-centric
Coordination	Autonomous	Orchestrated under Harmonia
Justification	Optional logging	Mandatory, auditable
Failure mode	Agent errors cascade	Human remains in control

---

9. Implementation Status

9.1 Current State

This document represents the **conceptual specification** (v0.1). It is:

- Architecturally complete
- Internally consistent
- Implementation-agnostic
- Awaiting reference implementation

9.2 Roadmap

Phase	Status	Description
Specification	Complete	This document
JSON Schemas	Complete	Memory and SenseMaking schemas
Reference Implementation	Planned	Minimal kernel in TypeScript
Solid Integration	Planned	Pod-based memory storage

Phase	Status	Description
Federation Protocol	Designed	P2P cluster communication

---

10. Conclusion

LifeOS is not a product to be purchased or a service to be subscribed to. It is a **specification for cognitive sovereignty** — a blueprint for building AI systems that serve humans rather than extract from them.

The architecture presented here — Kernel, MemoryOS, SenseMaking, Harmonia, Double Lock, Federation — forms a coherent whole. Each component reinforces the others. Together, they define a system where:

- Presence is respected
- Memory serves wisdom
- Interpretation remains human
- Action requires justification
- Sovereignty is preserved

This specification is released under MIT license. Anyone may implement it. No one may own it.

`"The right action is the one that needs no correction."`

References

1. Berners-Lee, T. (2017). Solid: A Platform for Decentralized Social Applications. MIT CSAIL.
2. LifeOS Specification Repository: https://github.com/ivan-berlocher/lifeos-spec
3. Solid Protocol: https://solidproject.org/

Appendix A: JSON Schema References

The complete JSON schemas for MemoryItem and SenseMakingFrame are available in the specification repository:

- `schemas/memory-item.json`
- `schemas/sense-making-frame.json`

Appendix B: Complete Example

A full trace of an intention through all LifeOS layers is provided in:

- `examples/intent-to-action.md`

This example demonstrates how a simple intention (“send an email”) is processed through Presence detection, Memory retrieval, SenseMaking synthesis, Harmonia deliberation, Double Lock verification, and final execution — showing the system’s protective and sovereign-preserving behavior.

Document version: 0.1
Last updated: December 2024
DOI: 10.5281/zenodo.XXXXXXX (pending)