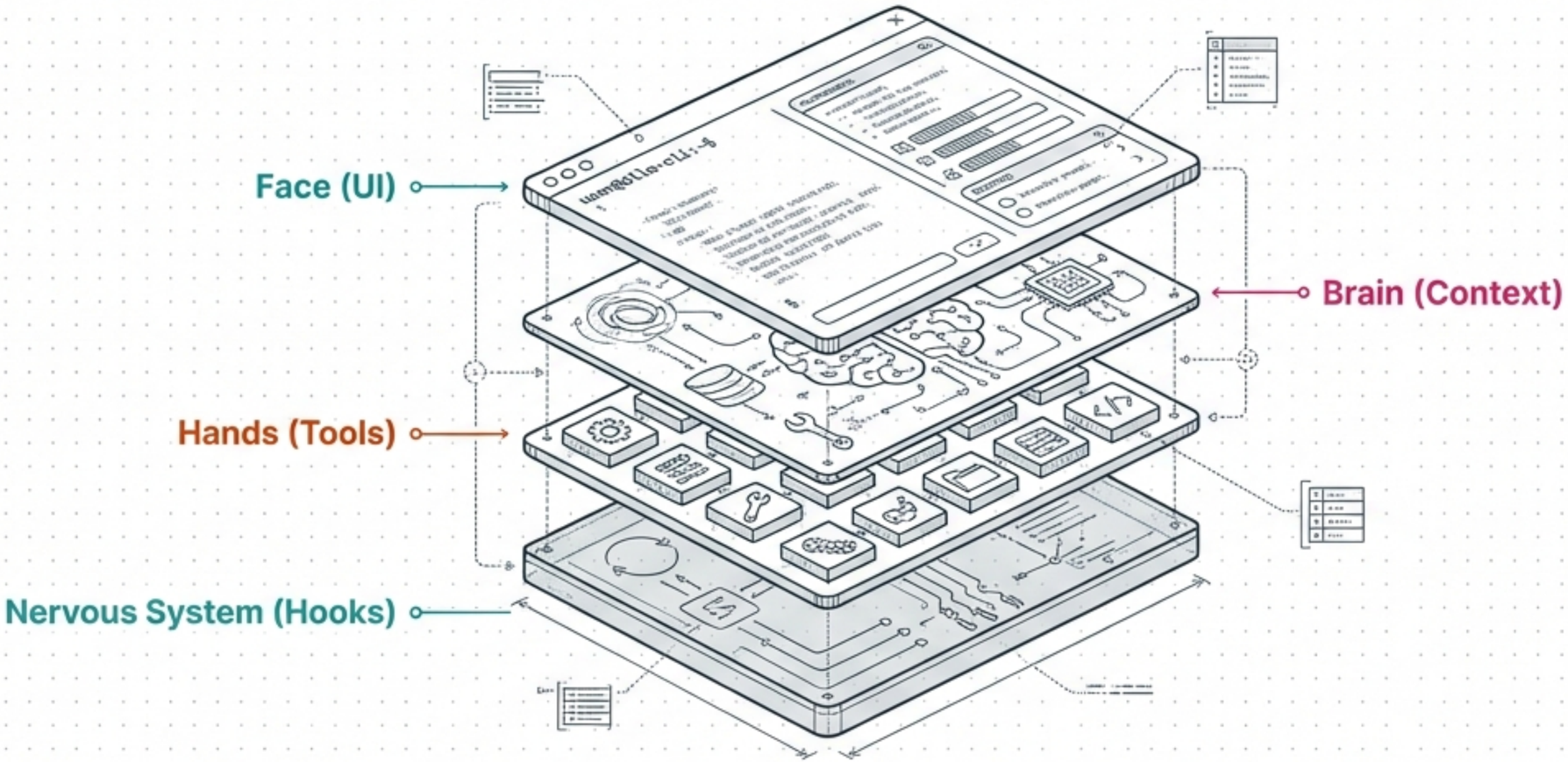
OLLM CLI: The Anatomy of Local Intelligence

A Technical Deep Dive into **Terminal-First AI Architecture.**



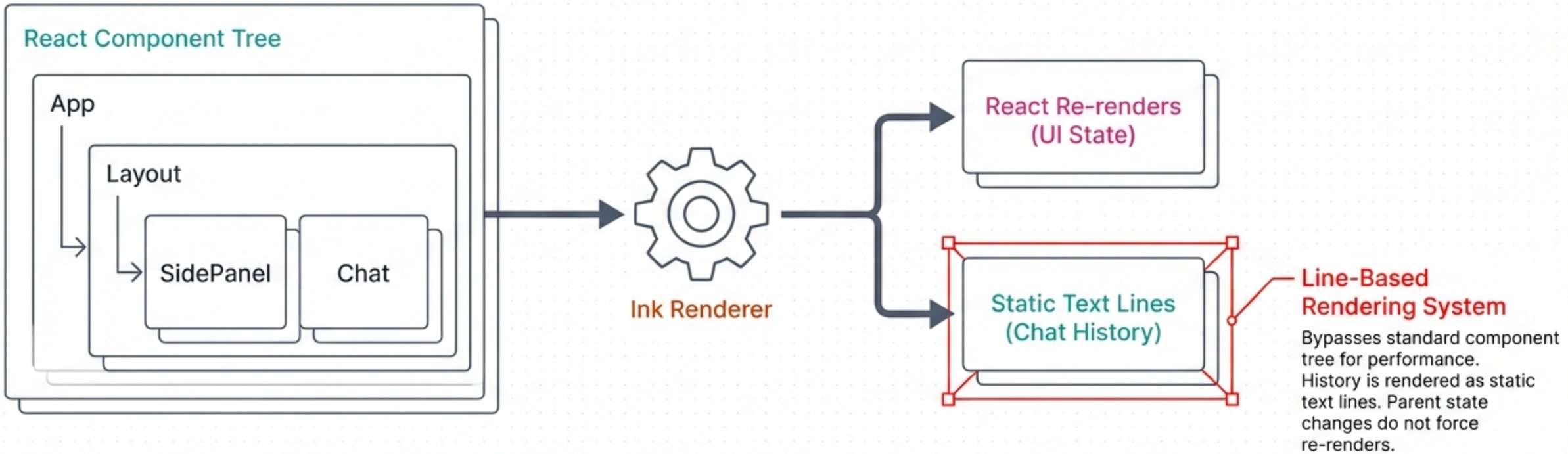
Local-First

Keyboard-Driven

Complex Workflows

The Face: React + Ink Rendering Pipeline

Core Tech Stack: React 19.2.3, Ink 6.6.0, TypeScript

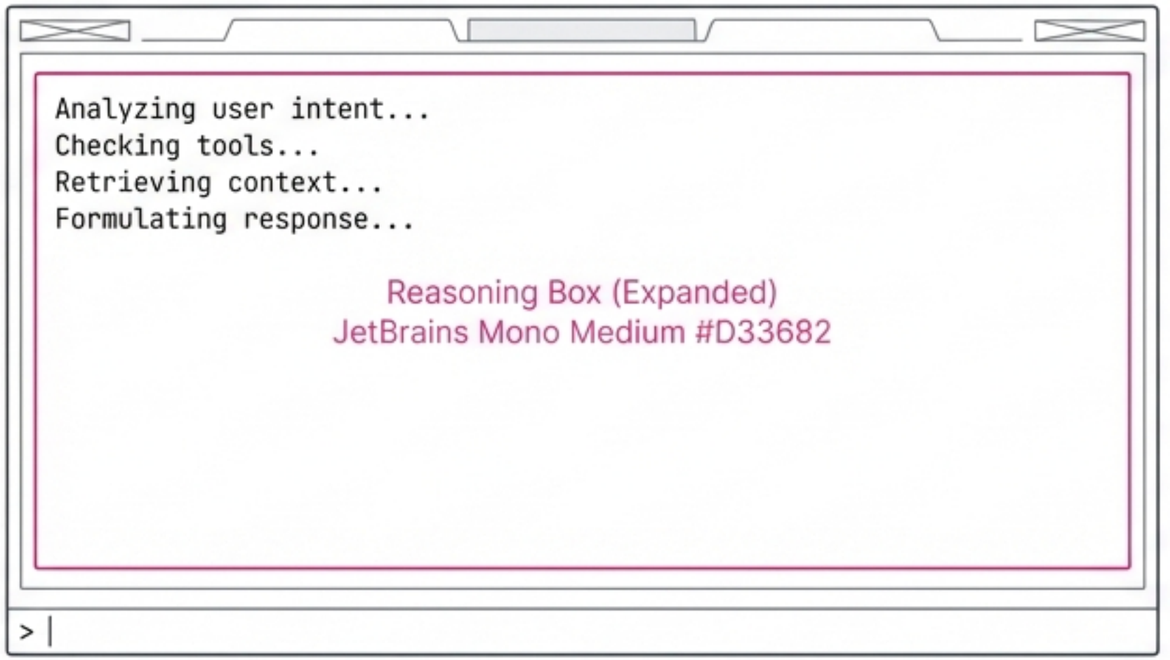


PTY Integration: Dual-terminal via **node-pty** **ANSI Serialisation**: **xterm.js** buffers to structured tokens **Rendering**: **RGB/Palette** color fidelity

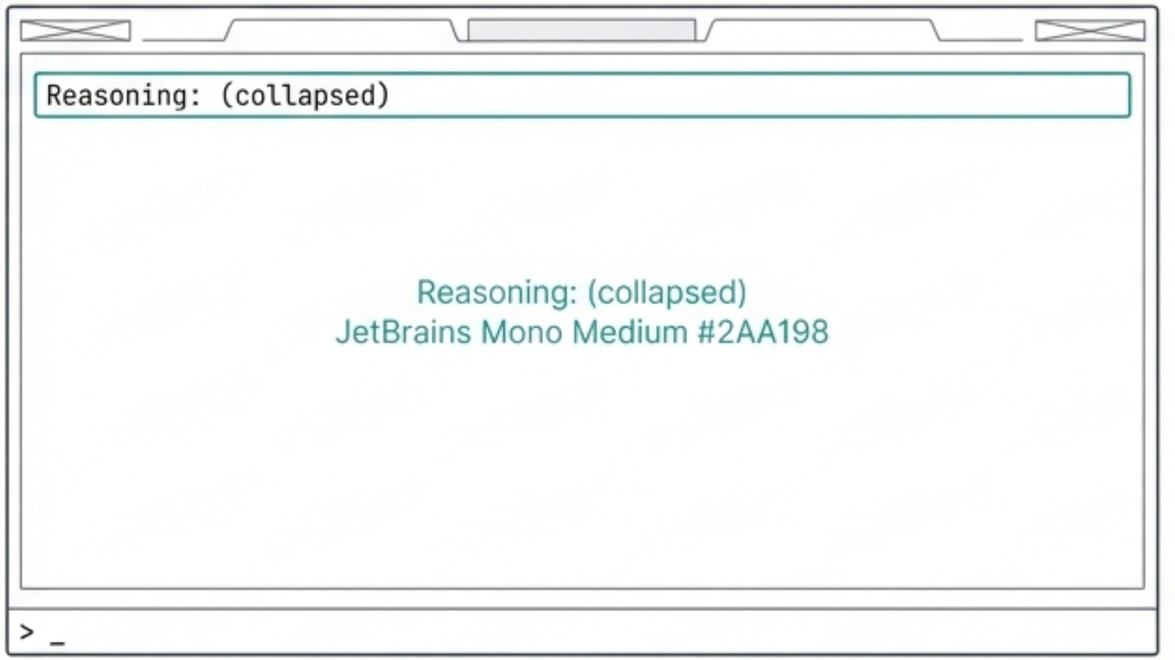
Visual Intelligence & Adaptive Modes

In Inter Bold Deep Slate #283238

State: Streaming



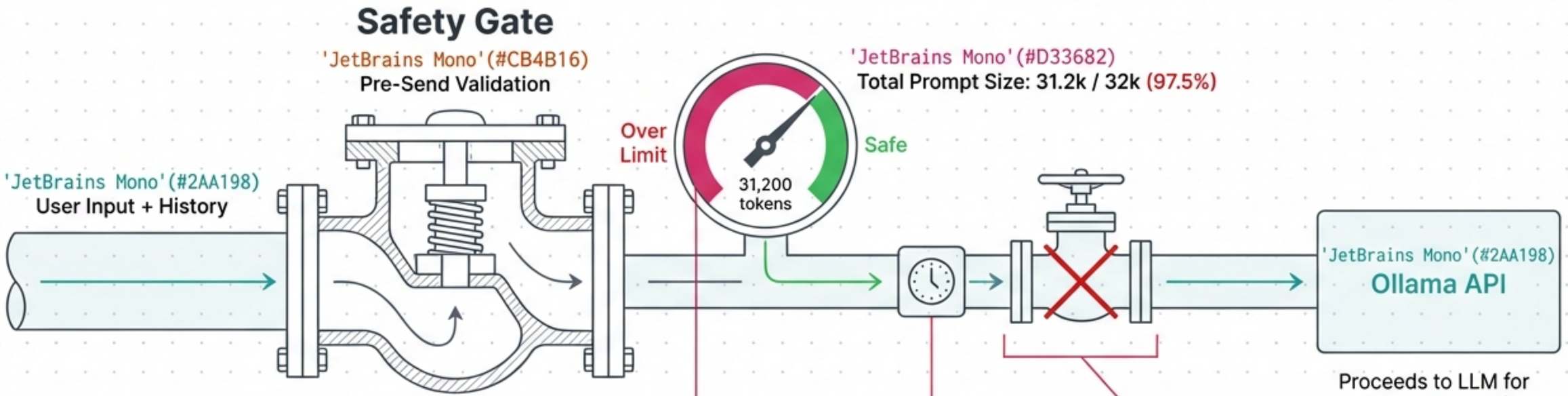
State: Complete



Reasoning Models	Context Telemetry	Focus Management
Native support for 'thinking' output (e.g., DeepSeek R1). Auto-expands during generation.	Real-time Header Bar visualization. **Green** <60%, **Yellow** <80%, **Red** >80%. Inter Regular Slate Grey	Keyboard-first navigation (No mouse). Vim-style alternatives. Inter Regular Slate Grey

The Brain: Context Management Strategy

Ensuring efficient data handling and preventing context window overflows with a proactive safety gate mechanism.



Context Management: Proactive safety gate, automatic compression, and race condition prevention. System integrity is paramount.

Cognitive Compression: The 3-Level Aging Strategy

Stage 1
Level 3: Detailed

Fresh Checkpoint.
500 Tokens.
Full summary + decisions.

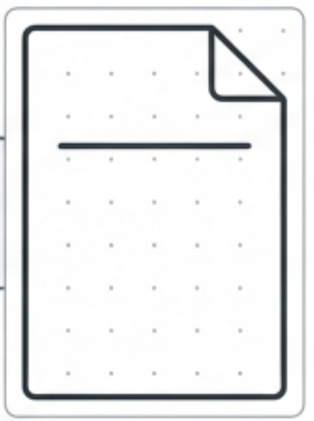
After 3 compressions

Stage 2
Level 2: Moderate

Key Decisions Only.
200 Tokens.

After 6 compressions

Stage 3
Level 1: Compact



One-line Summary.
50 Tokens.
“Ancient” history.

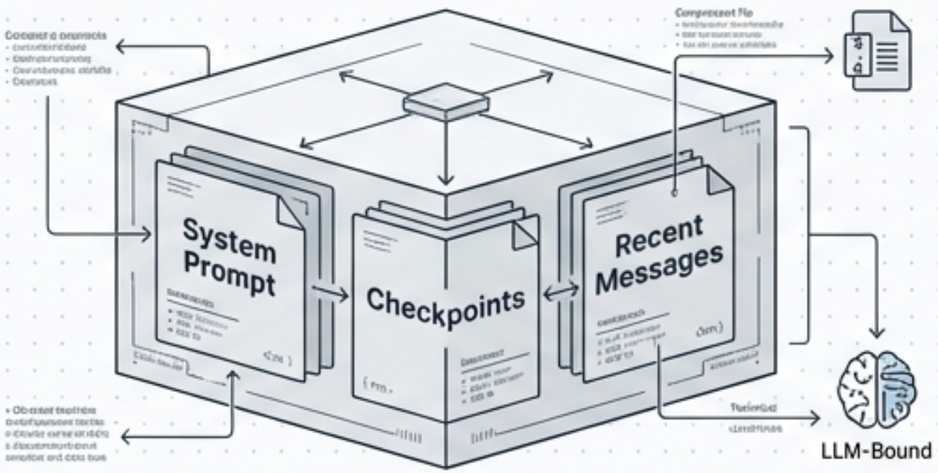
The Immune Layer (Never Compressed)

Goals
System Prompts
Active Tasks

Storage Layers: Volatile vs. Permanent

RAM

Active Context
(The Synapse)



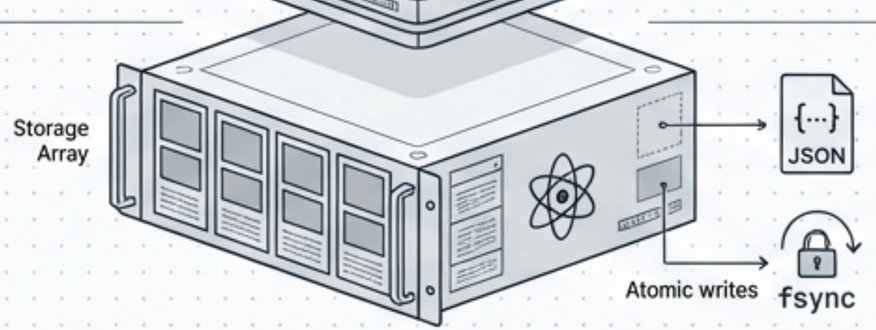
Highly compressed. LLM-Bound.
Contains System Prompt + Checkpoints + Recent Messages.

DISK

Snapshots
(The Safety Net)
`~/.ollm/context-snapshots/`

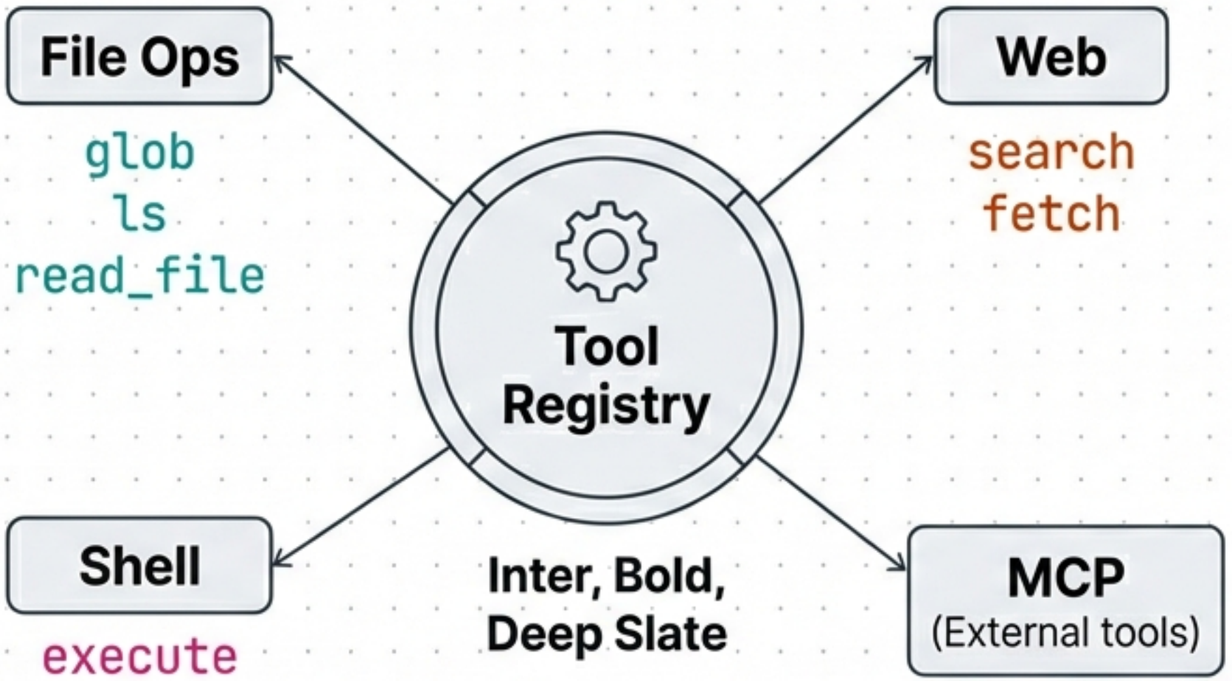
Point-in-time recovery.
Created automatically at 85% usage.
Allows Rollback.

Sessions
(The Black Box)
`~/.ollm/sessions/`



Full uncompressed history (JSON).
Atomic writes with 'fsync' ensure zero data loss.

The Hands: Unified Tool Execution System



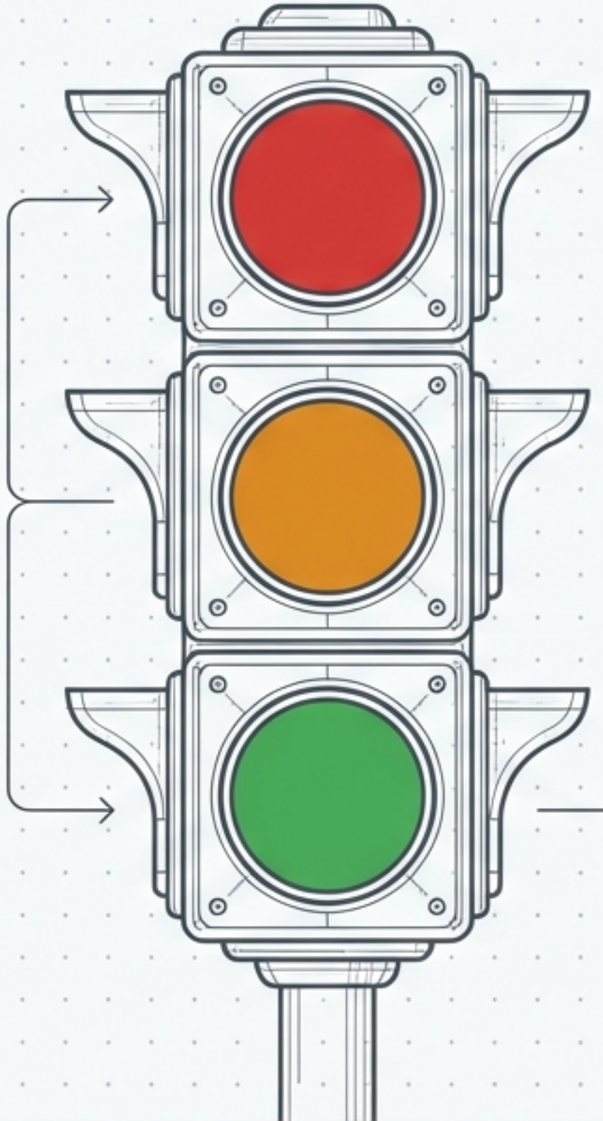
The Registry Architecture

- **Polymorphism:** Handles Built-in, Dynamic, and MCP tools identically.
- **Validation:** JSON Schema validation for all parameters.

Smart Formatting & Safety

- **Truncation:** File contents `>10KB` are snipped.
- **Isolation:** `try-catch` blocks prevent tool failures from crashing the core.

The Security Model: Policy & Permissions



ASK Mode (Maximum Safety)

Requires manual confirmation for EVERY action. Default for sensitive environments.

AUTO Mode (Balanced)

Read/Search/Memory = Auto-Approve.
Write/Shell/Edit = Require Confirmation.

e.g., `read_file`, `glob`, `search` (Cyan #2AA198) | `write_file`, `execute`, `edit_code` (#CB4B16)

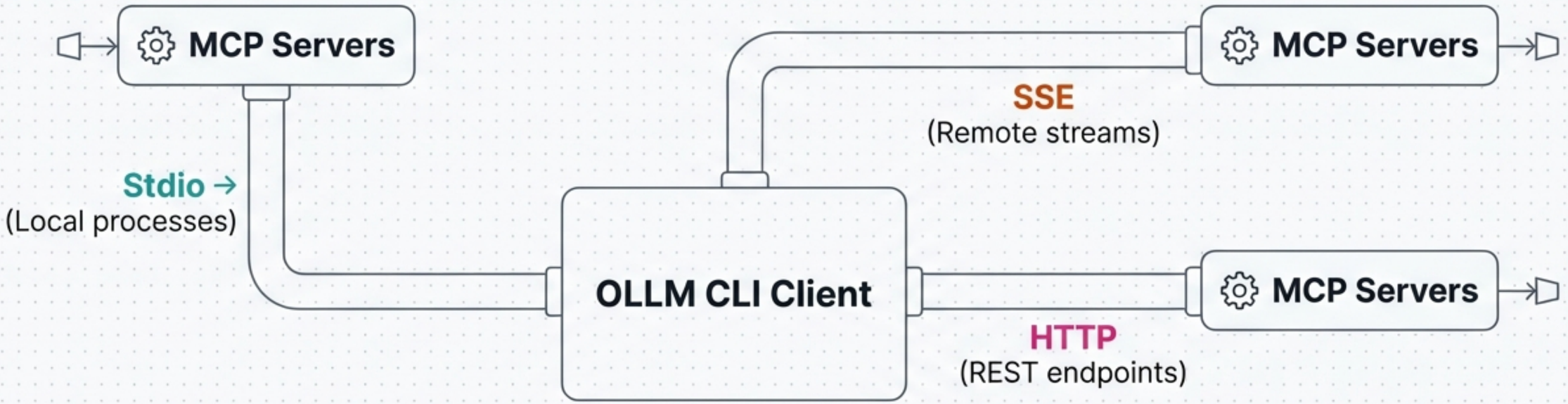
YOLO Mode (Maximum Speed)

Auto-approves all tools. No interruptions.

Sandboxing

- **Granular Permissions:** Extensions must declare specific needs (Filesystem, Network, Shell) enforced at runtime.
- Examples: `<permission name="filesystem.read" path="/home/user" />` (Cyan #2AA198)

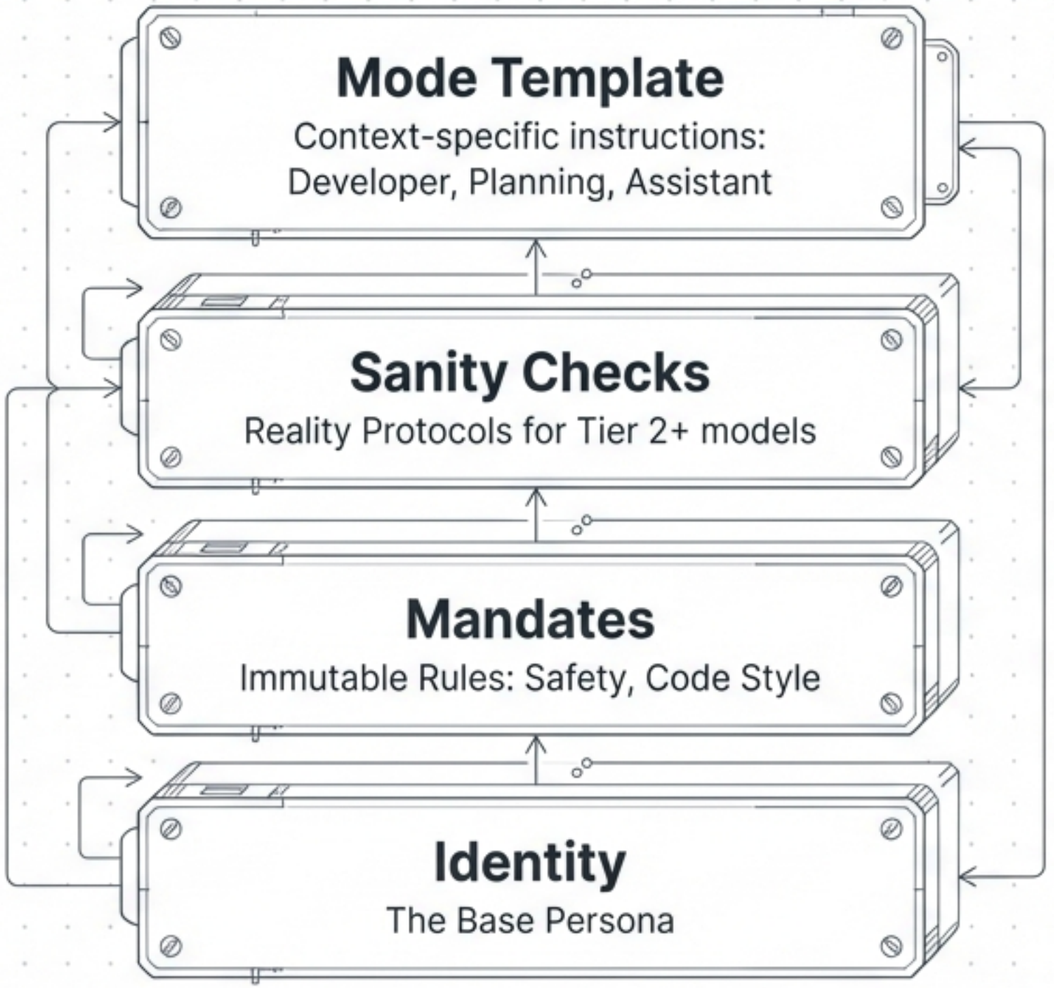
Extensibility: Model Context Protocol (MCP)



Key Architecture details

1. **OAuth 2.0** Integration: Secure **PKCE flows** with system **keychain token storage**.
2. Health Monitoring: Automatic background checks (**30s interval**) with **exponential backoff restarts**.
3. Ecosystem: Connects to any standard **MCP server** for unlimited tool expansion.

The Nervous System: Dynamic Prompt Engineering

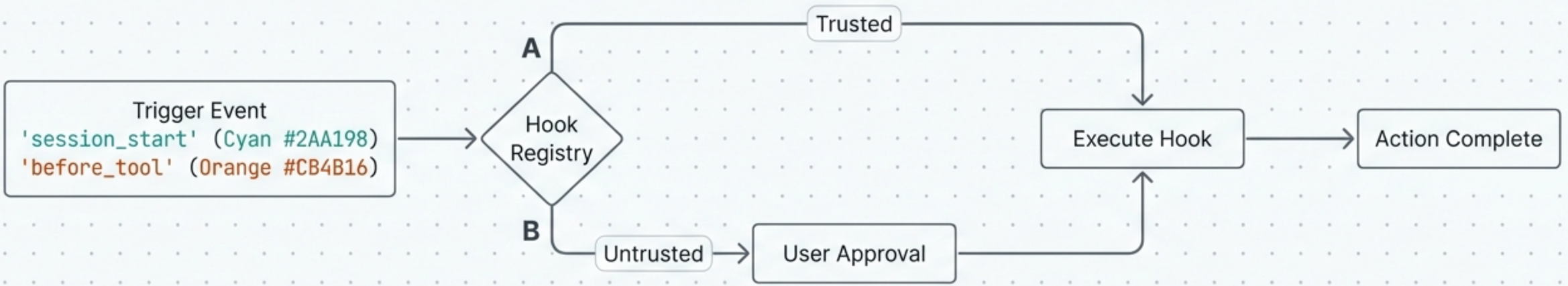


Tiered Intelligence

Prompt complexity scales automatically with context size:

- **Tier 1 (2K tokens)**: ~400 token prompt overhead.
- **Tier 5 (128K tokens)**: ~1800 token ultra-detailed instructions.

Event-Driven Architecture: The Hook System

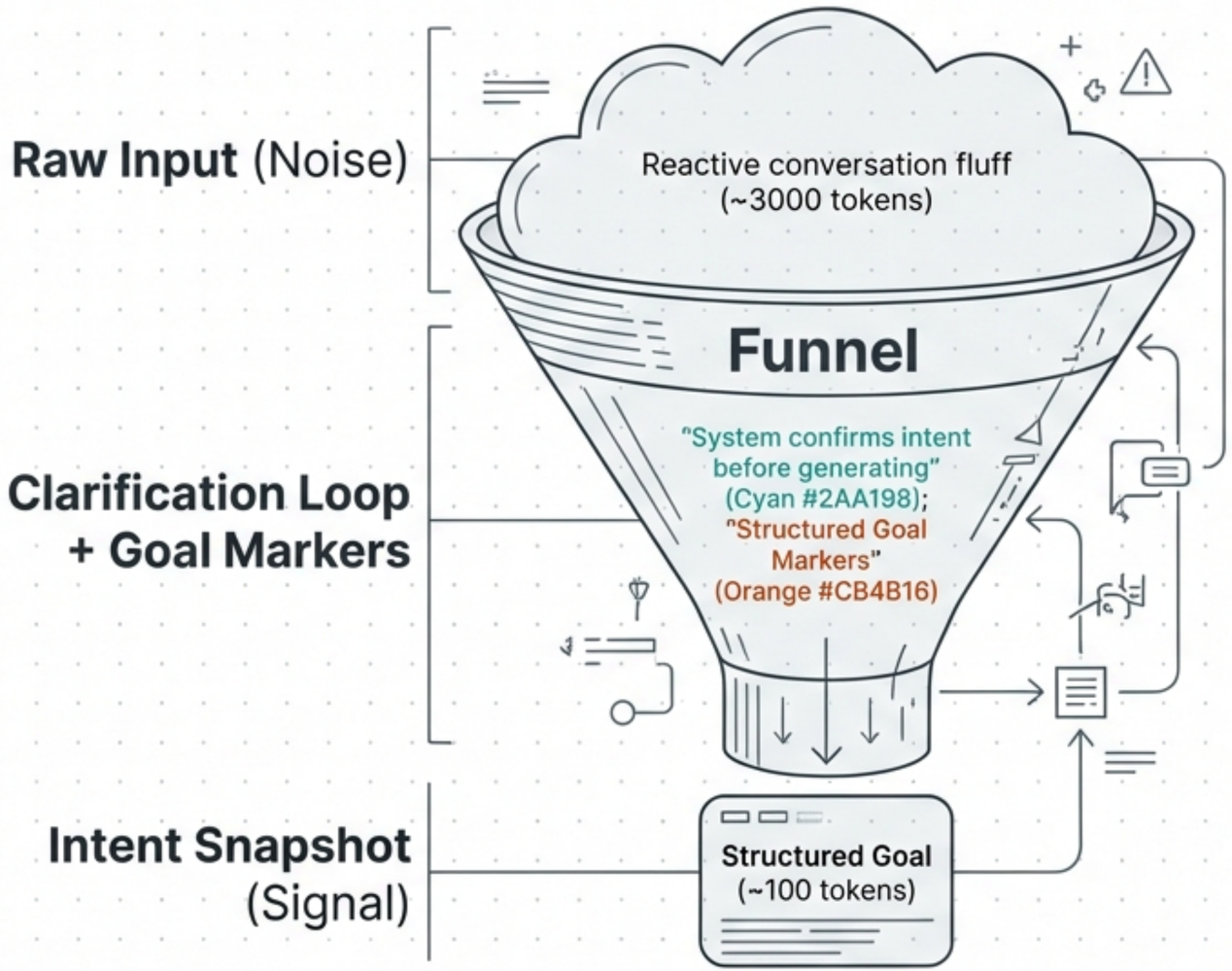


Key Features

1. **12 Lifecycle Events:** Including 'after_model' (Cyan #2AA198), 'pre_compress' (Orange #CB4B16), and 'before_tool' (Magenta #D33682).
2. **Trust Model:** Hash verification prevents malicious script tampering.
3. **Execution Strategy:** Supports both Sequential (ordered) and Parallel execution.

Input Preprocessing & Goal Extraction

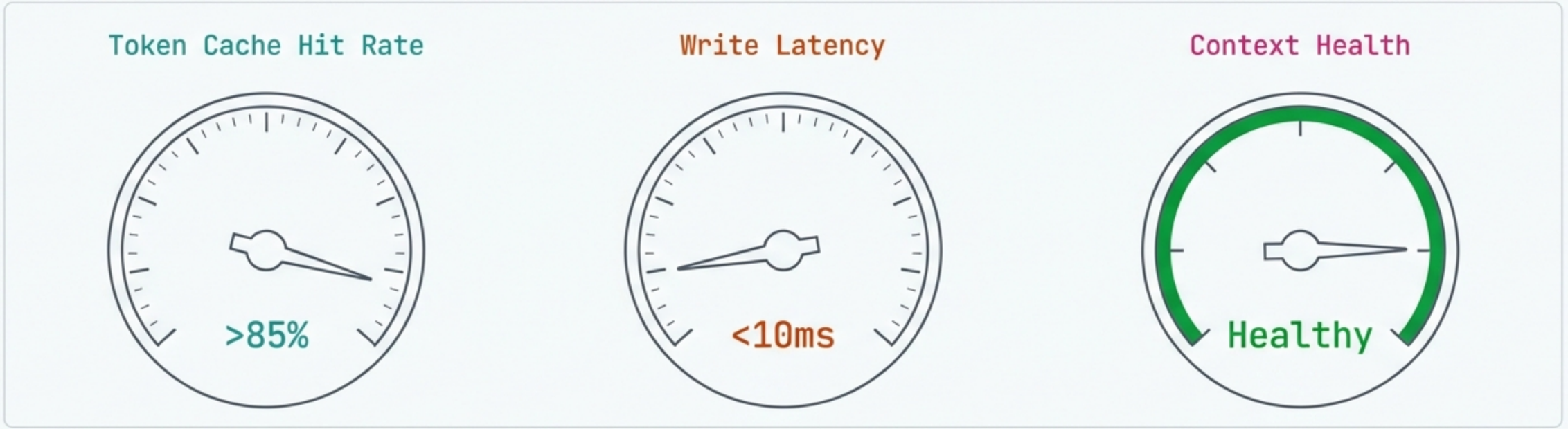
In 'Inter', Bold, Deep Slate (#283238)



The Solution: Proactive Goal Setting in 'Inter', Bold, Deep Slate

- **Clarification Loop: System confirms intent** before generating. 'JetBrains Mono' Cyan #2AA198)
- **Efficiency**: Reduces active context usage by up to 97% **(30x token savings)**. 'JetBrains Mono', Medium.

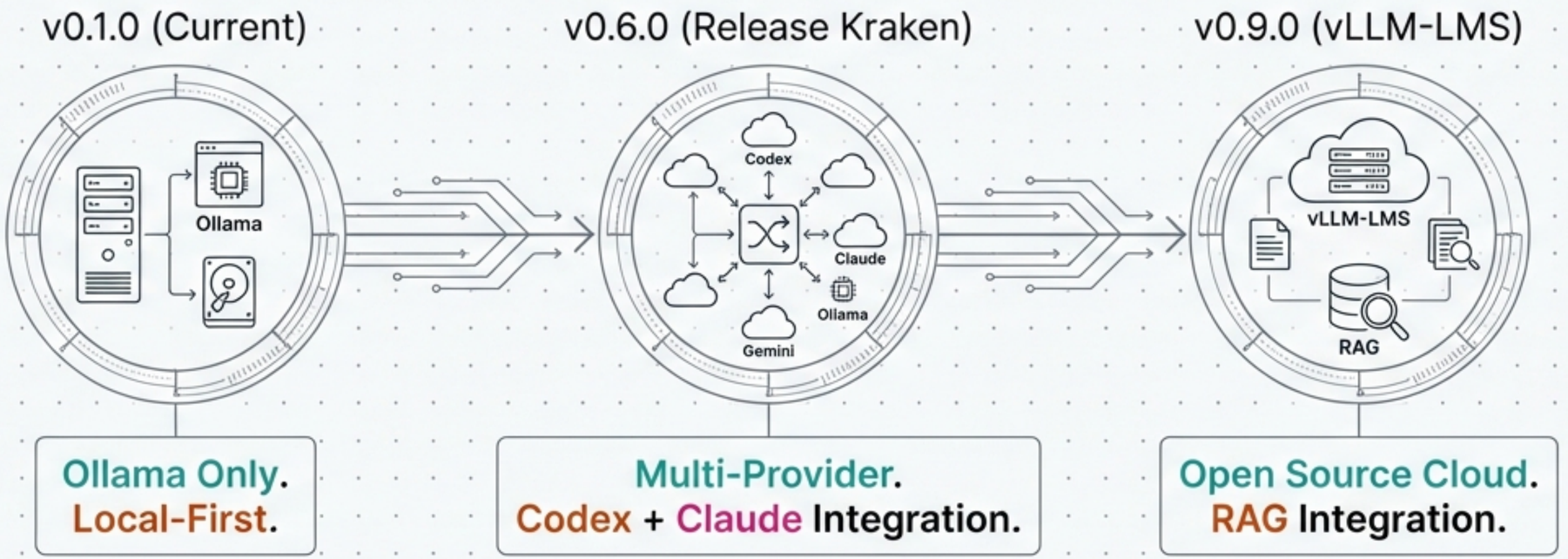
System Resilience & Reliability Metrics



- **Drift Detection:** Active warning system if calculated tokens diverge from tracked usage.
- **Storage Efficiency:** Atomic writes with `fsync` ensure data integrity.
- **Provider Abstraction:** Architecture ready for 'Release Kraken' (v0.6.0) supporting Codex/Claude/Gemini.

Future Architecture: Beyond Local-Only

Roadmap Timeline



Upcoming Capabilities

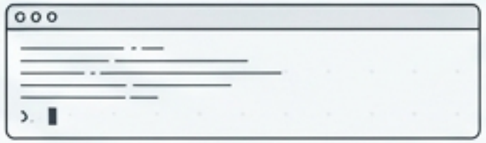
- Seamless Provider Switching: Local vs. Cloud backends.
- Reasoning Analytics: Metrics for "thinking" quality.
- RAG Integration: Turn Session Files into searchable knowledge.

The System Anatomy: Summary

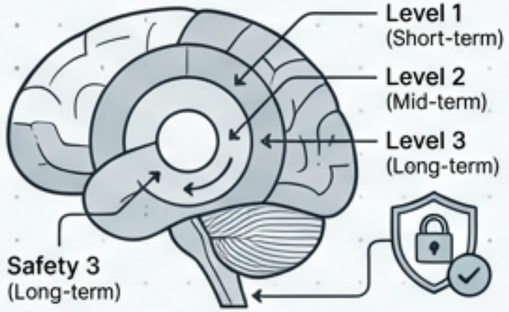
The Face (UI)

React (#2AA198) + Ink (#2AA198)

Line-Based Rendering.



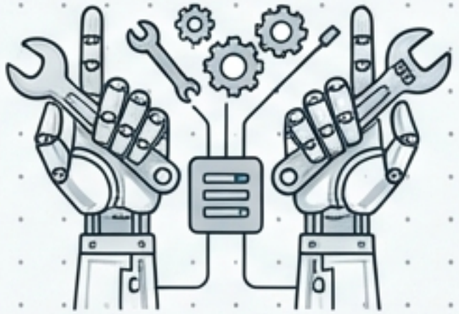
The Brain (Context)



3-Level Aging.

Safety Gate Validation.

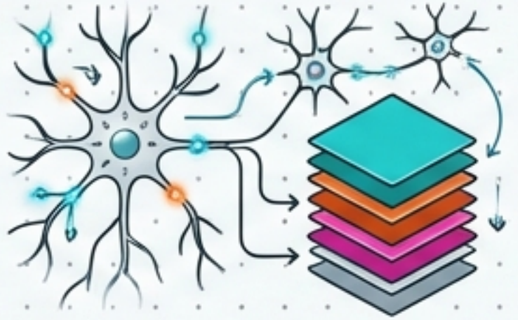
The Hands (Tools)



Unified Registry.

MCP + Policy Engine

The Nervous System (Hooks)



Event-Driven.

Dynamic Prompt Stack.

Final Takeaway: A production-ready, local-first AI architecture that sacrifices neither safety nor capability.