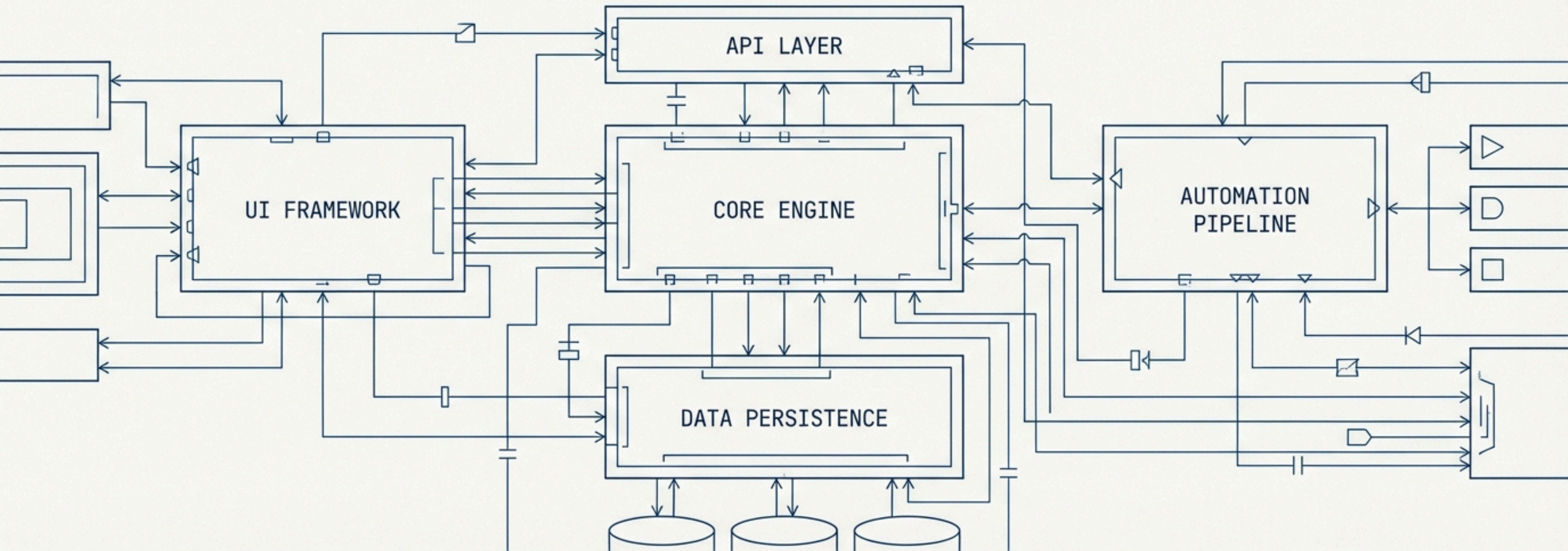


Engineering Complete Systems

A portfolio of work demonstrating a methodology for building production-grade platforms, engines, and toolkits for complex domains.



The Philosophy: Beyond the Application

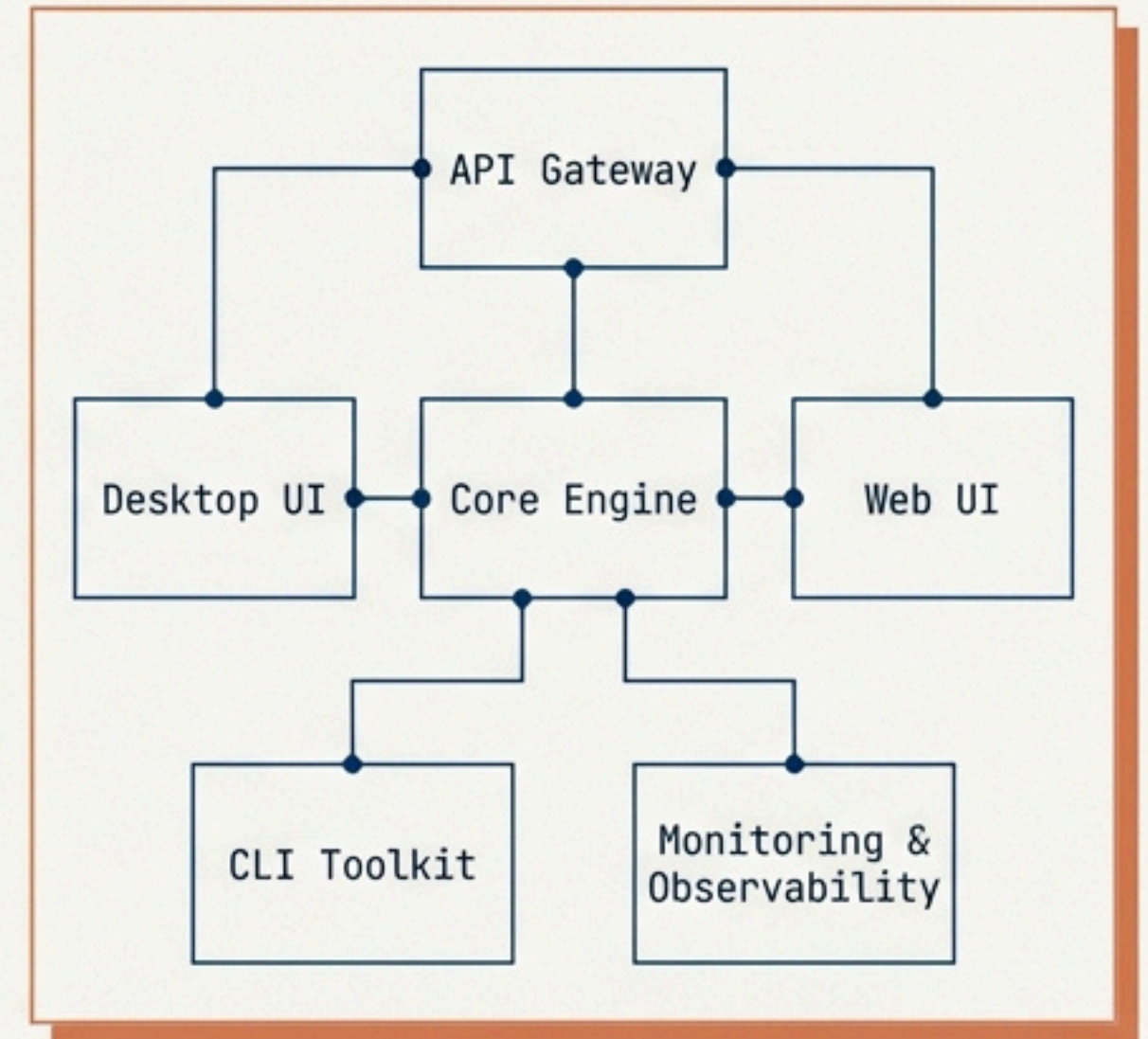
The work showcased here is guided by a single principle: move beyond isolated applications to engineer entire ecosystems. This means building systems characterized by:

- **Scientific & Technical Rigor:** Grounded in first principles, validated data, and robust engineering.
- **Enterprise-Grade Architecture:** Designed for scalability, security, and observability from day one.
- **Deep User Empowerment:** Providing multi-modal interfaces (GUI, CLI, API) that meet the needs of all users, from novice to expert.

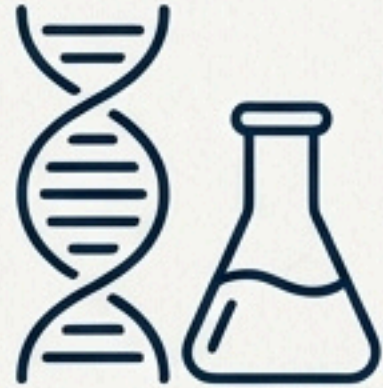
Application



Ecosystem



The Evidence: Four Proofs of a Unified Approach



1. N1o1

A Clinical Research & Pharmacokinetic Modeling OS.

Proof of: Scientific Rigor & Clinical Validation.



2. JustgettingReversed

A Universal Reverse Engineering Toolkit.

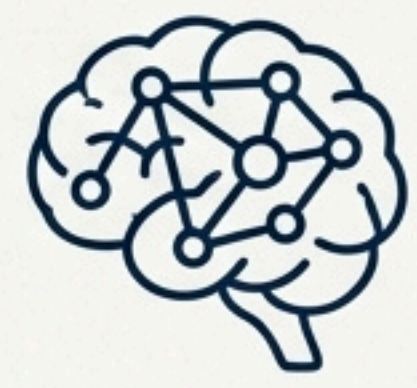
Proof of: Empowering All Users with Deep Tooling.



3. HHFE

An Enterprise OS for Scalar Wave Therapy.

Proof of: Enterprise-Grade Architecture for Niche Domains.



4. Cline

An Extensible AI Engineering Platform for VSCode.

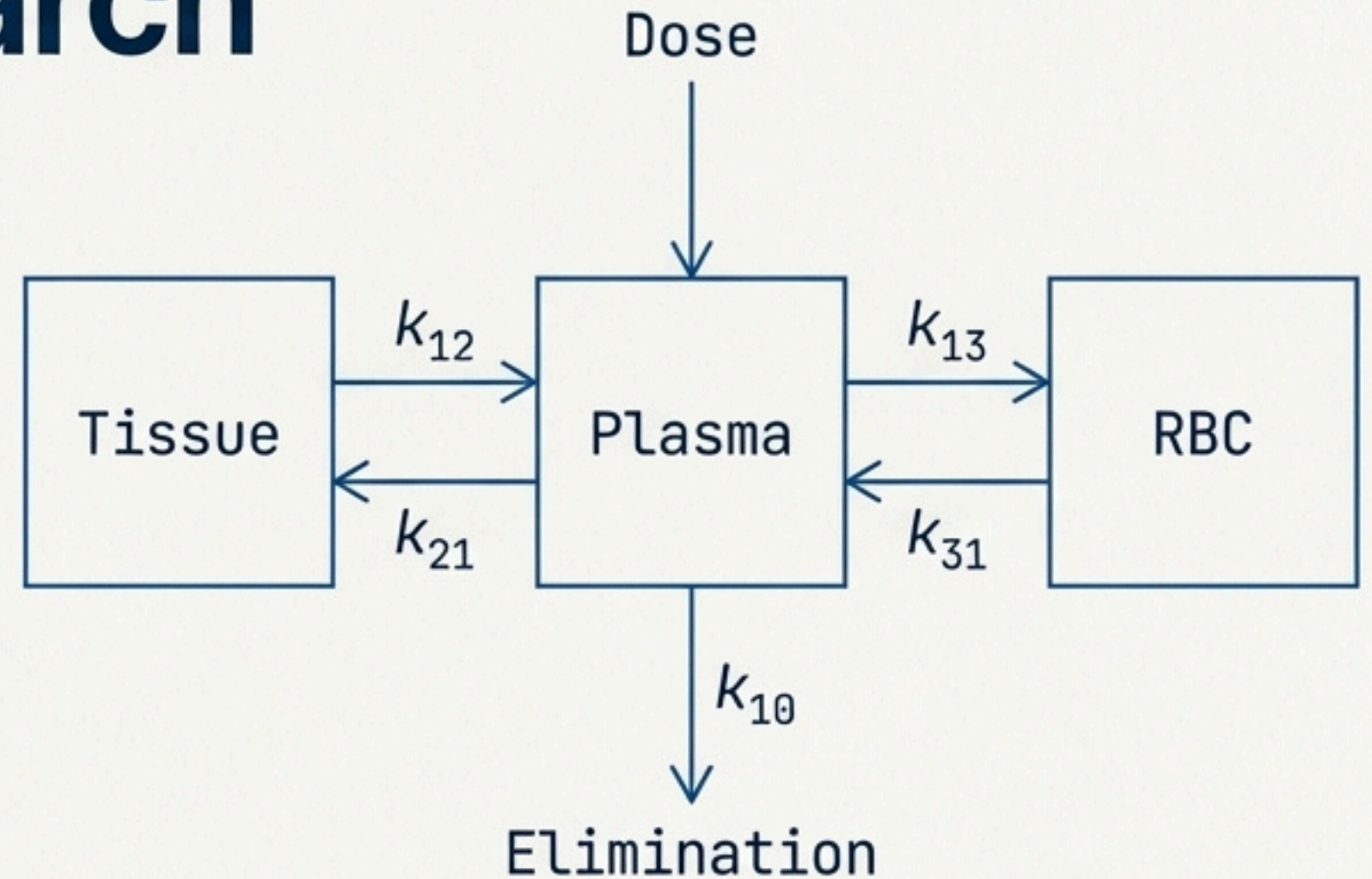
Proof of: Building an Extensible Platform for Experts.

Proof of Scientific Rigor & Validation

The frontier of non-invasive clinical research and pharmacokinetic modeling

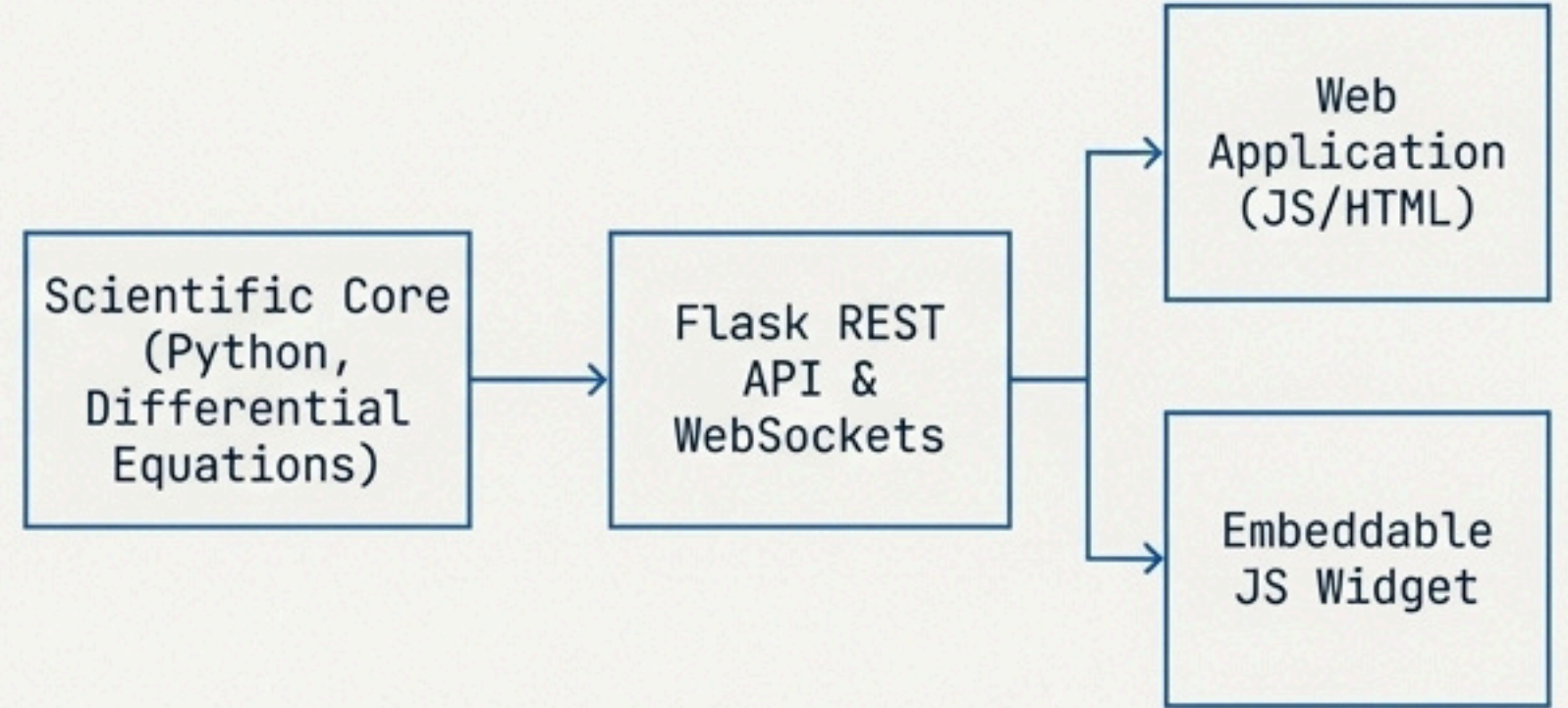
N1o1: A Clinical Research & Pharmacokinetic Modeling OS.

A complete platform for managing clinical trials, running complex biological simulations, and deploying validated health assessments to the public.



N1o1: Architecture & Proof

- Models complex biology with a multi-compartment pharmacokinetic model based on 10+ peer-reviewed papers (e.g., Lundberg et al., 2008; Kapil et al., 2010).
- Integrates a suite of AI-powered clinical tools for pre-screening, note generation, and sentiment analysis via a dedicated API (/api/ai-tools/).
- Delivers a full-featured REST API with WebSockets (/ws/monitoring) for real-time data streaming and management.
- Packages the core science into an embeddable, gamified "NO Score Widget" designed for mass adoption and engagement.



Clinically validated with an

$$R^2 = 0.87$$

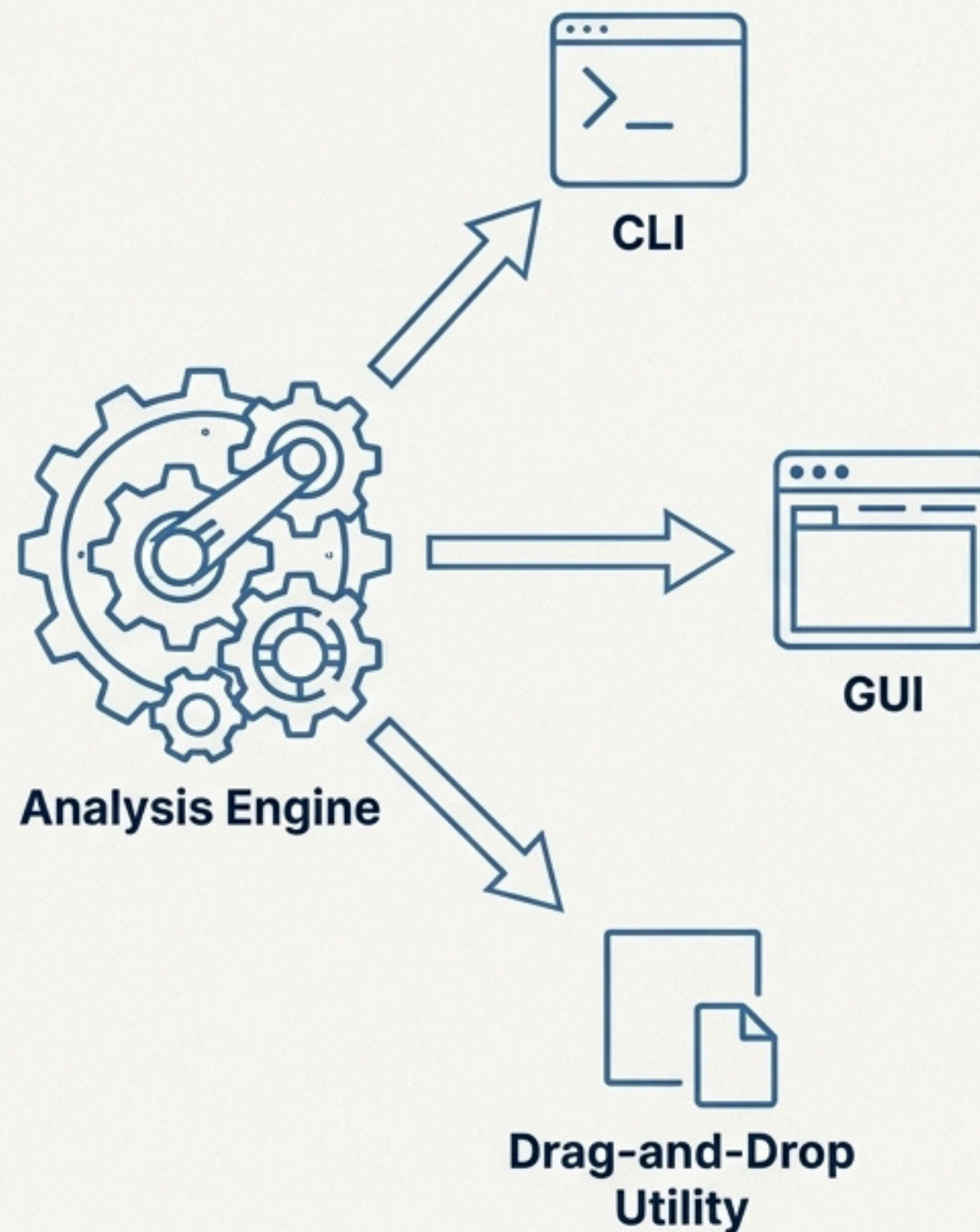
correlation against N=150
patient measurements.

Proof of Empowering Users with Deep Tooling

The intricate world of web application reverse engineering.

JustgettingReversed: A Universal Analysis Toolkit for Any User.

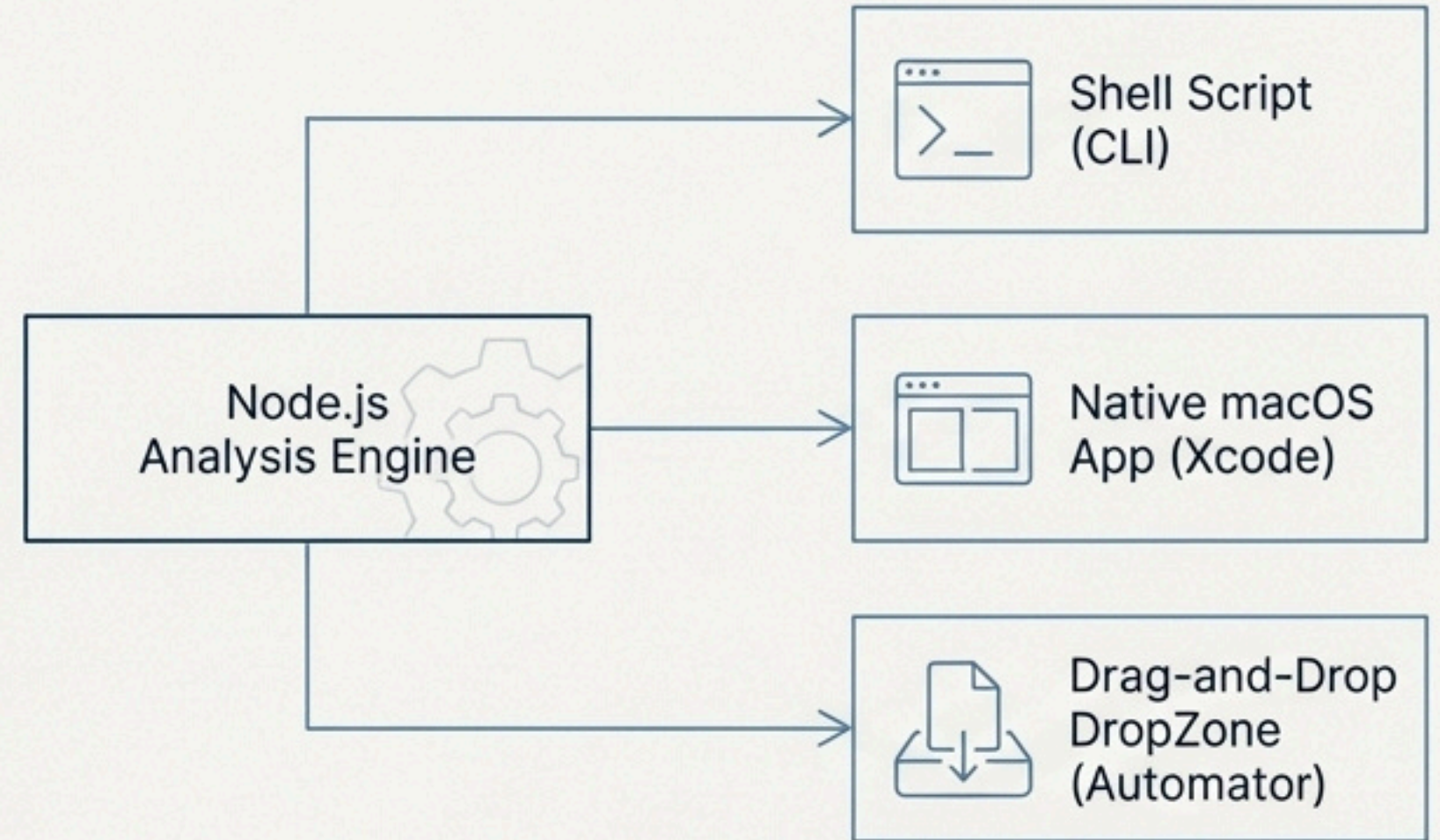
A complete, production-ready package that transforms a complex command-line engine into a universally accessible tool, regardless of the user's technical skill.



JustgettingReversed: Architecture & Proof

Key Capabilities

- * Engineered three distinct user interfaces to cater to different workflows: a scriptable CLI, a native GUI, and a simple drag-and-drop app.
- * Performs deep analysis of any JS application, capable of mapping 2,000+ API endpoints and identifying frameworks (React, Vue, Angular), security patterns, and auth flows.
- * Generates comprehensive, multi-format reports, including human-readable `ANALYSIS_SUMMARY.md` and machine-readable `universal-analysis.json`.
- * Systematically resolved a fragile build process, fixing hardcoded paths, adding error recovery, and creating a robust, distributable package.



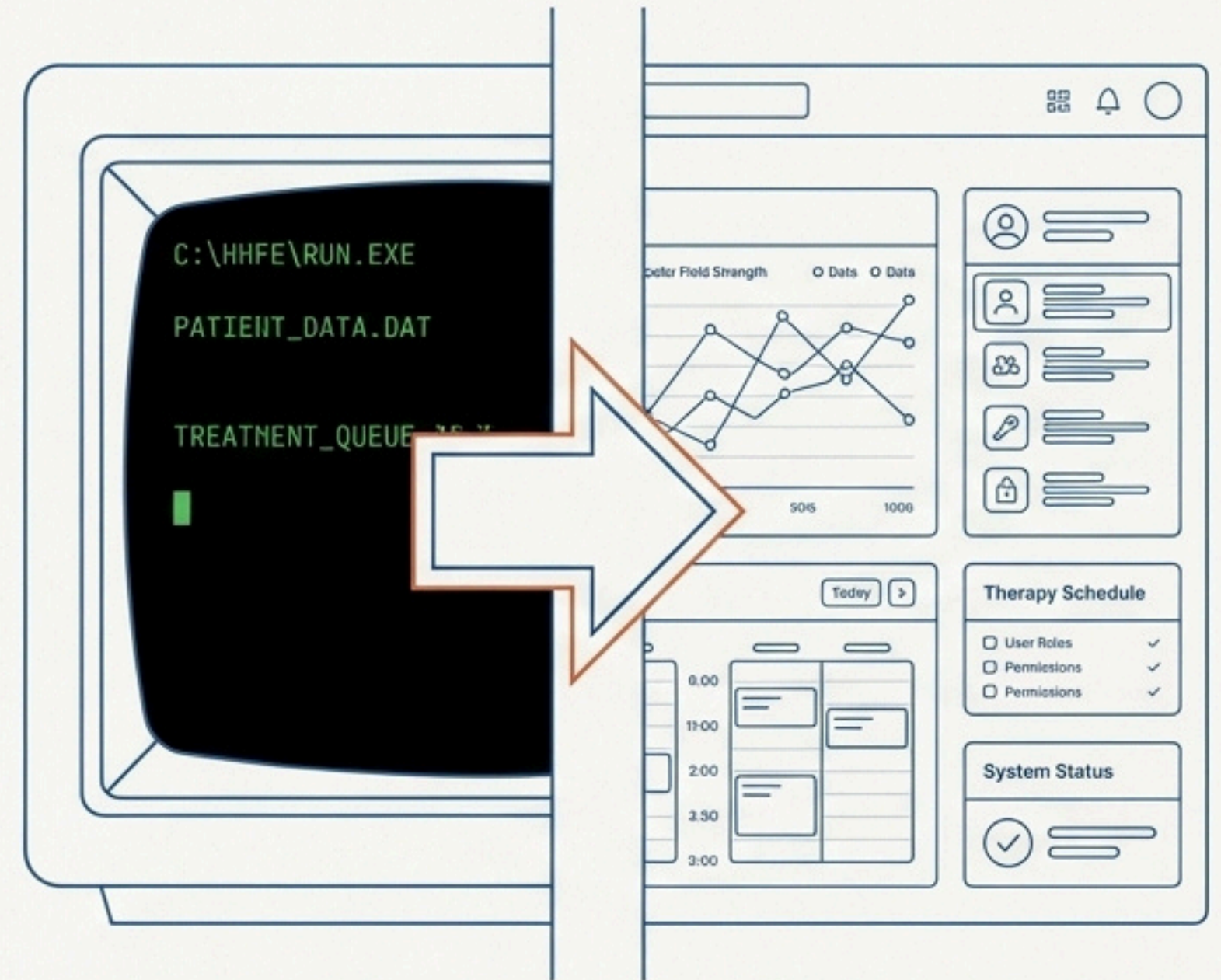
Achieved **Zero-Configuration** usability and a **100% build success rate** across all macOS setups, from a previously broken state.

Proof of Enterprise-Grade Architecture for Niche Domains

The emerging field of digital scalar wave therapy and data-driven wellness.

HHFE: An Enterprise OS for a Modern Wellness Center.

A full-stack platform for managing scalar wave therapy centers, modernized from a legacy DOS application into a secure, scalable, enterprise-ready system.



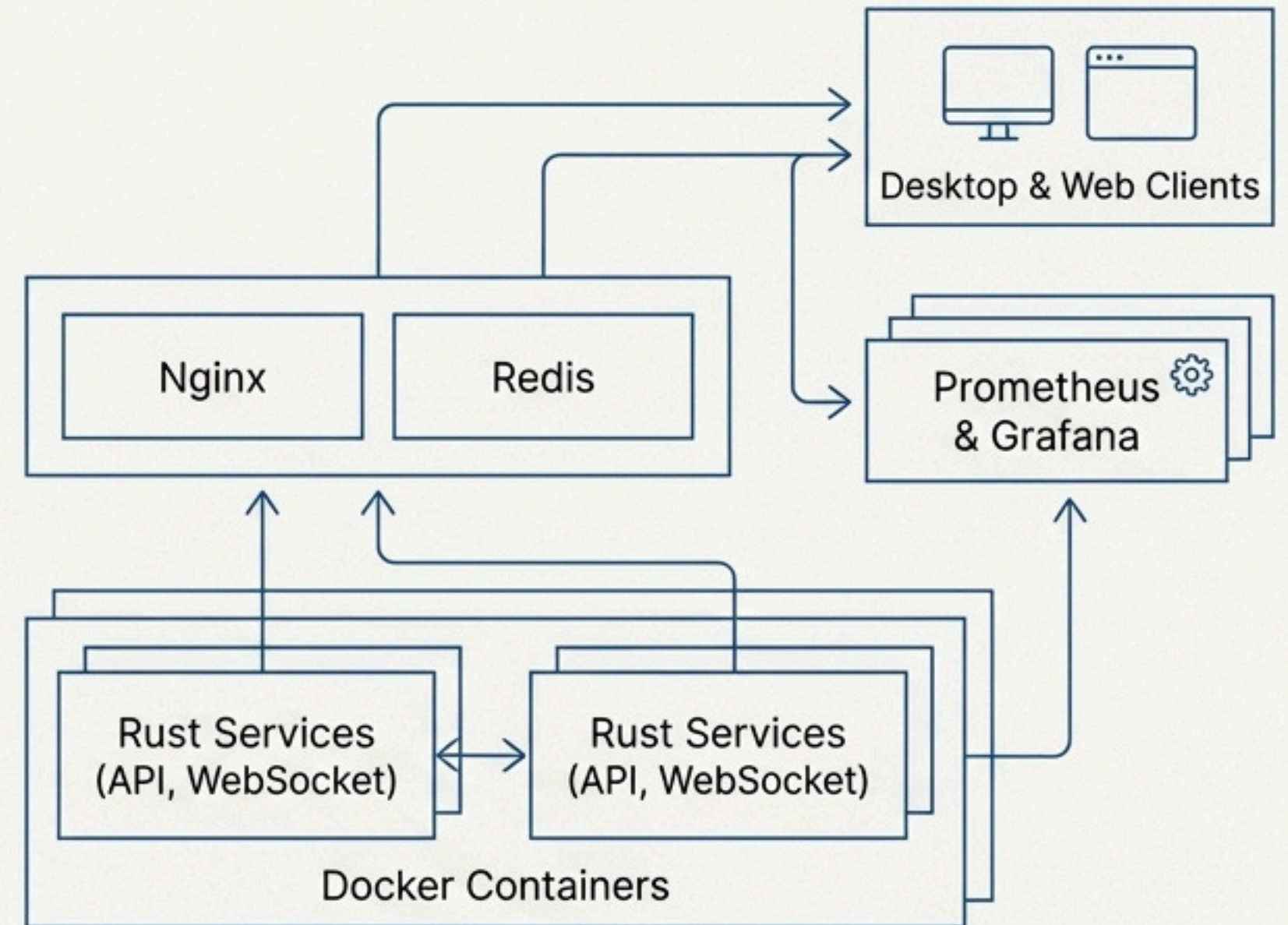
Legacy
JetBrains Mono

Modernized
JetBrains Mono

HHFE: Architecture & Proof

Key Capabilities

- * Executed a complete modernization of a legacy DOS application into a high-performance, memory-safe Rust architecture.
- * Designed as a modular Rust workspace with a core logic library (`hhfe-lib`), a desktop GUI (`hhfe-bin-mq`), and a scalable API server (`hhfe-api`).
- * Implements a full suite of enterprise features: JWT authentication, role-based access control (Admin, Practitioner, Client), comprehensive audit logging, and rate limiting.
- * Engineered for production observability from the start with integrated Prometheus metrics and Grafana dashboards for system monitoring.



Defining Metric: Applied a production-grade enterprise stack—**Rust, Docker, Prometheus, Grafana**—to a non-traditional domain, proving the methodology's versatility.

Proof of Building an Extensible Platform for Experts

The frontier of AI-native software development and agentic tooling.

Cline: An Extensible AI Engineering Platform for VSCode.

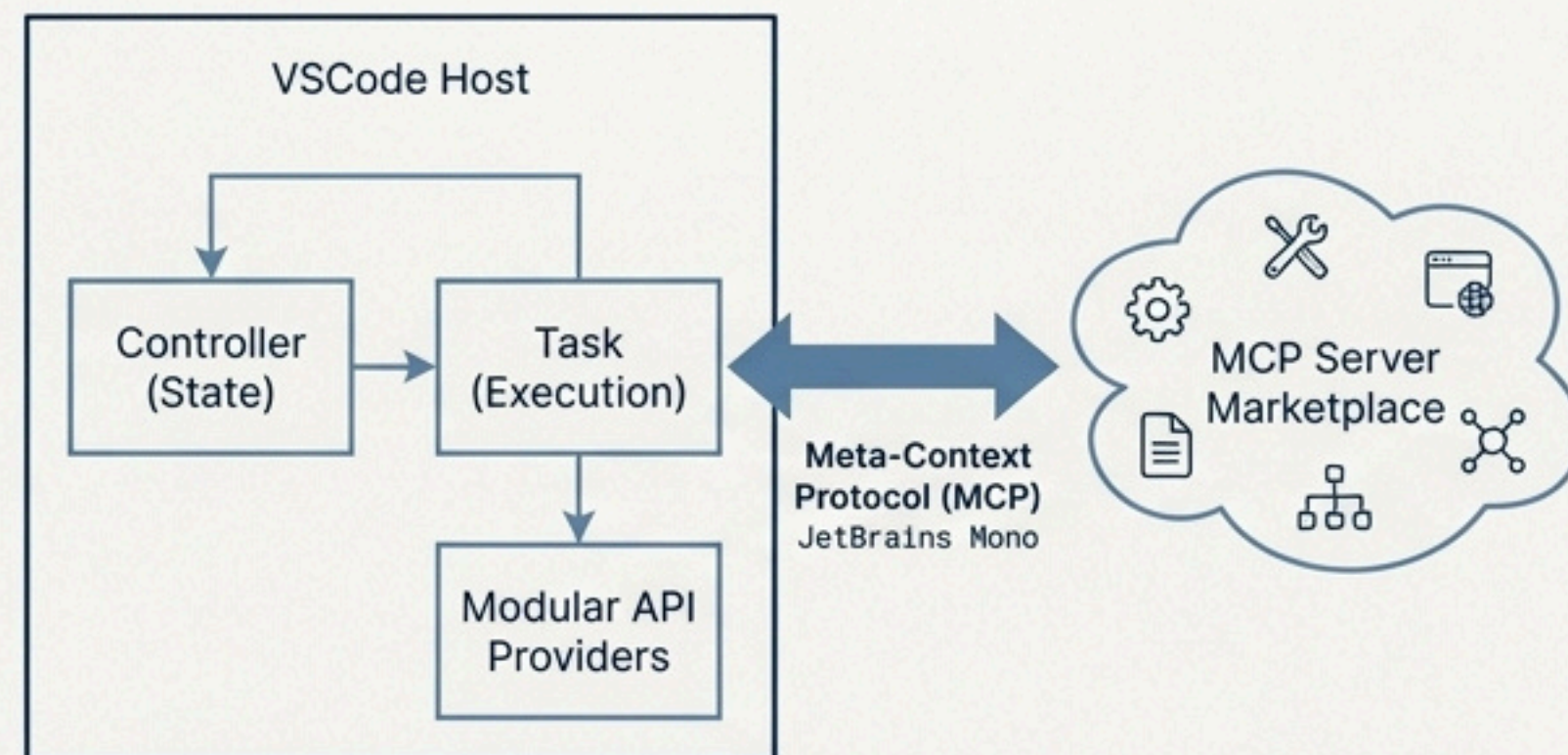
An AI coding assistant architected as an open platform, giving expert developers deep control over models, tools, and workflows within VSCode.



Cline: Architecture & Proof

Key Capabilities

- Built with a modular backend supporting diverse AI providers (Anthropic, Gemini, OpenRouter, local Ollama) and a versioned System Prompt Registry for fine-tuned AI behavior.
- Features an advanced Context Management System that intelligently preserves conversation history based on model-specific token limits (e.g., 200K for Claude, 1M for Gemini).
- Empowers expert users with persistent project knowledge via a "Memory Bank" (techContext.md, systemPatterns.md) and declarative instructions through version-controlled .clinerules.
- Designed as a true platform via the Meta-Context Protocol (MCP), allowing the community to build, share, and install new tools through the MCP Marketplace.



Defining Features: Engineered with a sophisticated **"Plan/Act" mode** for complex reasoning and an extensible **MCP Server architecture** for unlimited tool integration.

A Consistent Architectural Signature

Documentation as a Core Product	Pragmatic Modularity	Automation From Day One	Multi-Modal Interfaces
<ul style="list-style-type: none">• N1o1: SCIENTIFIC_BASIS.md, API_DOCUMENTATION.md• JGR: BUILD-INSTRUCTIONS.md, INSTALLATION-GUIDE.md• HHFE: ELEVATOR_PITCH.md, SECURITY.md• Cline: cline-memory-bank.mdx, mcp-server-development.mdx	<ul style="list-style-type: none">• N1o1: Clear separation of routes/, utils/, integrations/.• HHFE: Rust workspace with decoupled lib, bin, and api crates.• Cline: WebviewProvider -> Controller -> Task architecture.• JGR: Core analysis engine fully decoupled from its multiple UIs.	<ul style="list-style-type: none">• HHFE: Fully containerized with Docker; CI/CD workflows.• JGR: Automated build-and-fix scripts (build-fix.sh).• Cline: Comprehensive test suites (unit, e2e); snapshot testing for prompts.• N1o1: Extensive tests/directory validating all core scientific logic.	<ul style="list-style-type: none">• JGR: CLI, Native App, and Drag-and-Drop GUI.• N1o1: Full web app, embeddable widget, and REST API.• HHFE: Desktop GUI alongside a REST & WebSocket API.• Cline: Integrated VSCode UI plus extensible MCP servers.

A Proven Methodology for Future Challenges

This body of work is not a portfolio of disparate projects, but a demonstration of a repeatable methodology for architecting complete, production-grade systems in any domain. It proves the capability to tackle future challenges of any scale or complexity—from pioneering scientific research to building the next generation of AI-native platforms.

