

— AGENTIC WORKSPACE

Orbis Technical Brief

Agentic engineering workspace for ECU software, measurement analysis, and calibration drops with mandatory human review gates.

Early Access

A2L Search & Lookup Tools

Native PTY Terminal

Mandatory Human Review Gates

Generic AI assistants fail on automotive engineering.

Standard chat windows and coding copilots collapse when confronted with massive vehicle telemetry and million-line A2L descriptions.

CONTEXT EXHAUSTION

Multi-Gigabyte Logs

Generic LLMs cannot ingest 2GB+ MDF4 binaries or 2-million-line A2Ls into standard prompt context windows.

HALLUCINATED LABELS

Plausible Inventions

Chat interfaces frequently invent plausible-sounding calibration labels that do not exist in the ECU's actual ASAP2 database.

NO EXECUTION HARNESS

No Live Terminal

Without an interactive execution environment, assistants cannot run build scripts, verify HiL suites, or inspect CLI output.

An agentic workspace built for vehicle engineering.

Orbis provides a local desktop workspace where AI agents query automotive drops through specialized domain tools with mandatory human review gates.

DOMAIN TOOLING

Sparse A2L & Log Querying

Orbis never dumps raw files into context. Tools like a2l_search locate labels via regex, and a2l_lookup retrieves record layouts and flashed values decoded to physical units.

HUMAN REVIEW GATES

Diff Approval Required

Every file modification, hex patch, and shell execution surfaces as a clear visual diff requiring engineer sign-off: allow once, allow session, or deny.

Early Access Maturity: Orbis is currently in early access. It operates with strict human supervision and does not act autonomously.

Native terminal execution and procedure skills.

NATIVE PTY TERMINAL

Real Execution, Not Simulation

- **VTE Behavior:** Runs compiler builds, HiL test suites, pytest runs, and conversion utilities in a real terminal environment.
- **Context Streaming:** Live console streams return directly into the reasoning transcript for immediate agent evaluation.
- **Bi-directional MCP:** Speaks Model Context Protocol in both directions—hosts tools and consumes MDX, Hexx, and HELIX tool servers.

ENGINEERING SKILLS

Traceable Procedures

- **Codified SOPs:** Loads written engineering procedures—such as WLTP comparison rules or release gate checklists.
- **Provenance Trails:** Transcripts explicitly record which skill fired on which turn and the exact tool outputs that supported conclusions.
- **Persistent Sessions:** Resumes past sessions with full tool trails, terminal logs, and approval records intact.

Local by construction across flexible model runtimes.

ARCHITECTURE LAYER	SUPPORTED OPTIONS	DATA BOUNDARY
Desktop Core	macOS, Windows, Linux native application	Strictly local file access, zero cloud telemetry
Local Inference	Ollama, LM Studio (DeepSeek, Llama, Qwen)	Fully air-gapped, zero external network requests
Cloud Endpoints (Optional)	Anthropic, OpenAI, Gemini, xAI, OpenRouter	Explicit user-configured API endpoints only
Tool Protocol	Model Context Protocol (MCP) host & client, ACP	Local loopback (stdio / 127.0.0.1 SSE)

What we do not claim.

Strict early access scope and engineering safeguards.

EARLY ACCESS STATUS

Early Access Product

Orbis is an early access workbench. It is designed to assist engineers with routine scripting, diffing, and analysis; it is not a finished autonomous engineering agent.

NO UNSUPERVISED WRITES

Mandatory Human Approvals

Orbis does not modify production calibration files or execute commands autonomously without explicit human approval at each gated step.

How to evaluate Orbis on your engineering drops.

STEP 1

Point to Local Drop

Mount a test folder containing calibration drops, A2L files, and scripts. Observe automated drop intake classification.

STEP 2

Test A2L Regex Query

Execute targeted label searches over large A2L databases without file-buffering latency or context saturation.

STEP 3

Review Diff Gate

Stage a script execution or file edit, review the generated visual diff, and test allow/deny approval permissions.

EARLY ACCESS PILOT

Request Early Access to Orbis

contact@vehiclelab.in · <https://vehiclelab.in/docs/orbis-brief>