

VASIC DIGITAL • 基于证据的作品集



Vasic Digital

原生 AI 软件工程 • 基于证据的作品集

Helix 产品系列、vasic-digital 实用工具舰队及 Server Factory 工具链——这些 AI 开发系统均以可信为设计核心，并基于共享的工程 Constitution 构建。

140+

代码库舰队

43

LLM 供应商

25

已配置的 LINUX 发行版

439

通过测试 • MAIL SERVER
FACTORY

1. 概览——先见全局

这是一个由vasic.digital与milosvasic.ru共同使用的统一作品集。它描述的并非零散的副业项目，而是经过精心设计的**产品矩阵**：**大型应用产品**构建于**数十个小型、解耦、独立测试的模块**之上，整体由共享的工程**Constitution**统领，并通过**反虚假质检**机制验证。这种结构正是其与众不同之处。多数作品集仅是已完成项目的清单；而这里呈现的，是一个完整的系统——每个产品均由经过验证的可复用组件组装而成，每个组件均遵循相同的不可妥协规则，每一项宣称的功能背后都有实证支撑。主导语言为**Go**，辅以Kotlin/KMP、TypeScript/React、Python、Swift及Shell，各取所长——**Go**用于高吞吐量服务与库，Kotlin用于资源配置与跨平台移动端，TypeScript用于类型化前端，Python则作为AI/ML的粘合层。

让整个作品体系凝聚一致的，是其纪律的机械性而非愿景性。共享的**Constitution**以Git子模块形式发布，并被140多个代码库继承，因此单一规则变更即可全局生效；反虚假质检层则拒绝在缺乏运行时证明的情况下记录通过。Helix产品系列、实用工具矩阵及Server Factory工具链，均是同一理念的三种表达——构建一次，复用无处不在，并证明其有效后方可宣告完成。

以下内容按优先级排序：

1. **治理与质检支柱**——HelixConstitution、HelixQA（确保其余部分可信的基石）。
2. **Helix产品系列**——AI开发生命周期（优先采用HelixTrack）。
3. **LLM基础设施**——供应商抽象、编排、验证。
4. **vasic-digital实用工具**——产品级独立工具。
5. **Server Factory**——基础设施自动化遗产（排序靠后）。

统一核心理念：**功能只有在真实用户能够使用且有实证证明时，方可视为完成。**

2. 治理与质检支柱

这两项居于首位，因为作品集集中的其他内容均借由它们获得可信度。二者共同将”相信我，它能用”转化为可审计的事实——**Constitution**编码规则，**HelixQA**证明规则被遵循。

- **HelixConstitution**——通用的、与项目无关的工程规则手册，以Git子模块形式发布，并被140多个代码库继承。具备反虚假验证门禁、假阳性免疫、数据/主机安全、覆盖率纪律；支持扩展但不可削弱的继承机制；全矩阵传播门禁通过grep检查必备条款；每个门禁均配有变异测试，以证明其并非摆设。
- **HelixQA**——反虚假质检编排（基于Go）。包含书面**YAML**测试库，以及全自动的LLM与计算机视觉质检流程，覆盖Android、Android TV、Web及桌面端；无实证（截图、logcat、视频、堆栈跟踪）则不予通过。**Constitution**强制要求的质检类型（§ 11.4.169）。

3. Helix 产品家族

Helix 系列是旗舰产品线：一个紧密衔接的产品家族，覆盖整个 AI 开发生命周期，从规划与需求定义，到构建、记忆、翻译及交付。每款产品均为独立实体，但设计初衷在于彼此组合——统一的治理框架、可复用的模块，以及贯穿所有产品的严谨验证机制。

- **HelixTrack** —— 面向自由世界的 JIRA 替代方案；Helix-Track 系列的旗舰产品。
- **HelixAgent** —— 集成式 LLM 服务：多模型协同决策，输出共识答案，并基于验证结果选择服务提供方。
- **HelixCode** —— 企业级分布式 AI 开发平台；将任务分配至 SSH 管理的工作节点，支持自动检查点/回滚；提供 REST、CLI、TUI 及 MCP 接口。
- **HelixCluster** —— 面向 AI 计算的分布式操作系统，从数据中心 GPU 到边缘手持设备，均由单一控制平面统一管理。
- **HelixBuilder** —— 基于 AI 的应用构建流水线，按类别逐步构建。
- **HelixSkills** —— 受治理且基于宪章的 CLI AI 代理技能系统（技能、MCP 工具服务器、Claude Code 插件）。
- **HelixSpecifier** —— 规模化的规约驱动开发，仪式感与工作量匹配。
- **HelixMemory** —— 面向 AI 代理的统一记忆大脑，融合四大顶尖引擎。
- **HelixTranslate** —— 经验证的模型书籍翻译；设计上诚实透明，绝不静默回退。
- **HelixTerminator** —— 零信任终端平台：每个 SSH 会话均受保护、可共享，并获得 AI 辅助。
- **HelixGitpx** —— 跨十余家主机的联邦 Git 系统；单一真相源，全局镜像同步。
- **HelixOTA** —— 通用的解耦空中更新机制；设计上确保零砖化风险。
- **HelixPlay** —— 将任意 GPU 设备转化为私有云游戏终端。
- **Helix-Flow** —— Helix 平台推理产品。未验证 / 因源码限制暂缓：其公开仓库目前仅有一行 README；待真实文档完善后再深入展示产品细节。

4. LLM 基础设施

产品之下，是确保其与提供方无关且可靠运行的基础层：统一接口整合数十家 LLM 提供商，单一控制平面管理无界面编码代理，以及唯一的验证真相源。正是这一层，使上层产品能够灵活切换模型、应对提供方宕机，并拒绝信任无法证明理解任务的 LLM 服务。

- **HelixLLM** —— 单一二进制，六种模式：支持 OpenAI 及 Anthropic 兼容推理（基于 HTTP/3、本地 llama.cpp、评分式回退链、RAG 流水线、ReAct 代理）。
- **LLMProvider** —— 单一接口，43 家提供商，内置断路器、重试机制及健康检查。
- **LLMOrchestrator** —— 面向所有无界面 CLI 编码代理（OpenCode、Claude Code、Gemini、Junie、Qwen Code）的统一控制平面。
- **LLMsVerifier** —— 验证、监控、优化：LLM/提供商/验证元数据的唯一真相源，并强制要求模型理解任务的准入门槛。

内容

5. Vasic-Digital 实用工具

这些独立的产品级工具各自解决了一个棘手的难题——同时也在实战中验证了可复用模块库的能力。它们涵盖了从弹性多协议媒体系统到 Markdown 转视频课程流水线，再到内容哈希化文档/数据库同步引擎等多种工具；部分工具坦诚公布了其成熟度，这些标识均保留而非隐藏。

- **Catalogizer** —— 支持多协议（SMB/FTP/NFS/WebDAV/本地）、加密、可自托管的媒体库管理工具；基于 Go/Gin API + React UI；具备弹性离线容错监控功能；构建于 21 个 `digital.vasic.*` 子模块之上。
- **Courses-Creator** —— Markdown 转视频课程流水线；支持多 LLM 内容增强、TTS（Bark/SpeechT5）、桌面/移动/网页播放器；优雅兼容无 API 密钥模式。
- **VisionEngine** —— 解耦的 Go 工具包，融合经典计算机视觉与多供应商 LLM 视觉技术；支持导航图生成（BFS + DOT/JSON/Mermaid 导出）；OpenCV 构建标签门控。
- **DocProcessor** —— 文档到功能映射工具，附带验证覆盖率追踪；支持 LLM 或启发式/离线提取；基于 Apache-2.0 协议。
- **Docs Chain** —— 内容哈希化、双向、原子化文档/数据库同步（基于 DAG 的 Salsa 风格增量重算）。阶段 1-5 已就绪；阶段 6-7 规划中。
- **Herald** —— 可靠的多渠道通知系统，具备自然语言三层意图解析（命令 → LLM → 确认）；首个 Docs Chain 消费者。
- **task_bridge** —— 解耦的双向任务/看板同步（SQLite 单一真相源 ? 文档 ? ClickUp）。P1 阶段搭建完成——同步逻辑尚未实现。
- **Vasic Digital 可复用模块套件** —— `digital.vasic.*` 「标准库」：基础设施原语、AI 构建模块及防御性 LLM 防护机制，外加 Kotlin Multiplatform 镜像。多个组织仓库处于搭建/开发中——已标识，未发布。

6. Server Factory（基础设施自动化传承——按设计排末位）

按设计而非质量排末位：Server Factory 工具链早于 AI 系列诞生，展示了「一次构建，随处复用」理念的最初形态。其旗舰产品——一款可通过 JSON 描述并部署于任意环境的邮件服务器——是一款成熟且经过充分验证的产品；其余配套工具则按实际成熟度呈现，而非刻意包装。

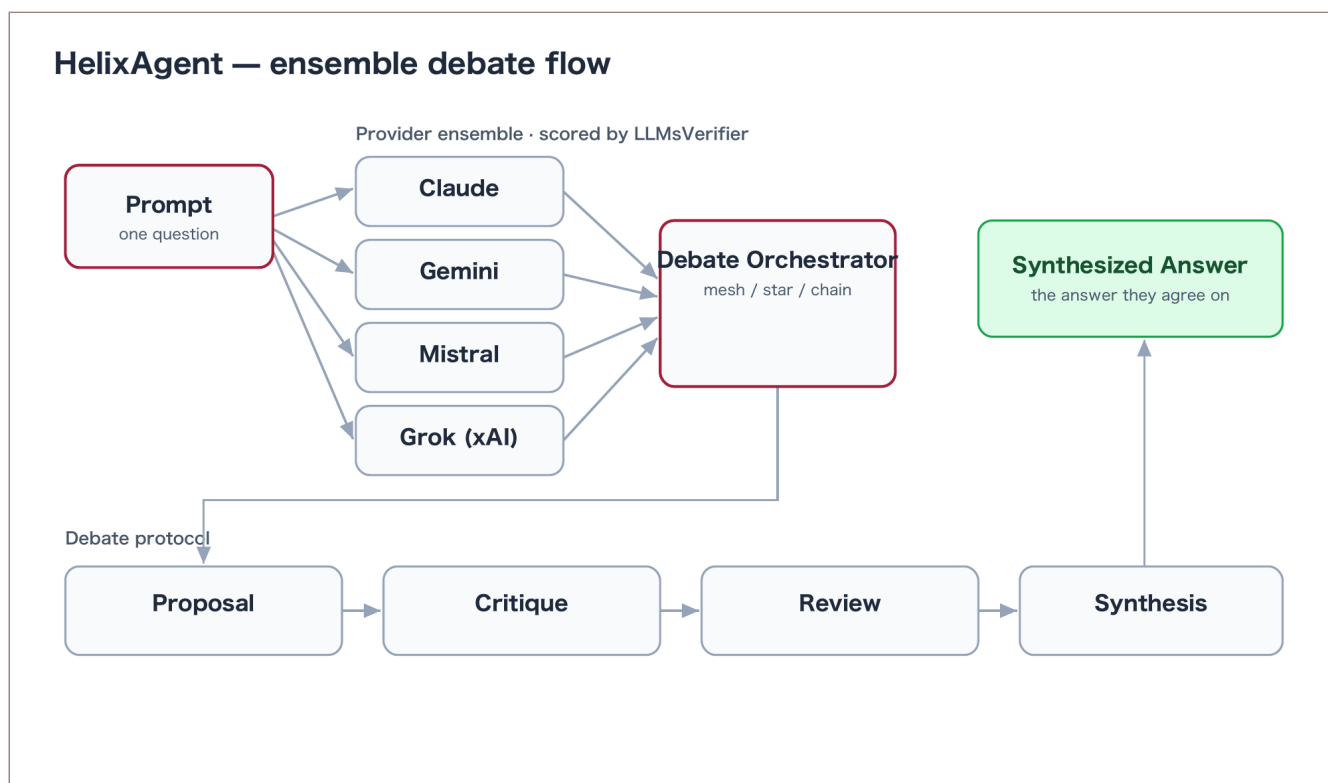
- **Mail Server Factory** —— 通过声明式 JSON 生成完全预配置的 Docker 化邮件服务器，支持 12 种连接类型及 25 种 Linux 发行版；具备企业级安全性；已通过 439 项测试及 SonarQube 门禁检查。Server-Factory 组织的旗舰产品。
- **Server Factory 核心框架** —— 所有工厂构建所依赖的共享 Kotlin 引擎。
- **Qemu-Utills** —— 将 QEMU 虚拟机镜像视作制品管理：下载/缓存/运行、压缩/发布、桥接/TAP 网络、ISO 安装；支持 Linux 及 macOS。
- **Parallels-Utills** —— Parallels（macOS）虚拟机镜像的压缩、发布及检索，通过简单配置文件实现。
- **Server Factory —— 附加组件** —— 服务工厂（Web/SonarQube/缓存代理）、定义包及实用工具。服务工厂文档为占位说明——未验证/早期阶段。

7. 技术指标（基于实证）

- **语言**：Go（主导）、Kotlin & Kotlin Multiplatform、TypeScript、Python、Swift、Shell；PL/pgSQL；TLA+（螺旋集群中的正式规格）。

- **AI / LLM:** 40+ 供应商接入、MCP、RAG、vector 数据库及 embeddings、规划（HiPlan/MCTS/思维树）、LLMOps、基准测试（SWE-bench/HumanEval/MMLU）、TTS（Bark/SpeechT5）、计算机视觉 + LLM 视觉、防护栏/红队测试。
- **后端:** Gin、gRPC + Protobuf、HTTP/3 (QUIC)、WebSockets、Angular、React、Kafka/RabbitMQ。
- **数据:** PostgreSQL、SQLite、SQLCipher、Redis、Neo4j、ClickHouse、MinIO/S3/GCS/Azure。
- **基础设施 / 运维:** Docker & Compose、Kubernetes + Helm、Prometheus + Grafana、OpenTelemetry、QEMU/Libvirt/Parallels; GitHub Actions、Gradle、Make。
- **测试 / 质量保证:** HelixQA、带变异门的挑战测试框架、`go test -race`、视觉回归测试工具、ADB 设备测试、SonarQube、安全扫描（semgrep/gosec/trivy/snyk/gitleaks/nancy）、TLA+ 模型检查。

精选架构



HelixAgent — 一种集成式 LLM 服务，供应商在四阶段协议下进行讨论，并交付达成共识的答案，由 LLMsVerifier 评分。

HelixCluster — seven-layer stack

14 control-plane
microservices

L7 · Federation & Observability

L6 · Security & Attestation (SPIFFE, PQ-E2EE)

L5 · Sessions & Interactive Terminal

L4 · AI Inference Routing

L3 · Omega Scheduler (two-level)

L2 · Consensus & Membership (Raft + SWIM)

L1 · Node Runtime & Transport (gRPC, WireGuard)

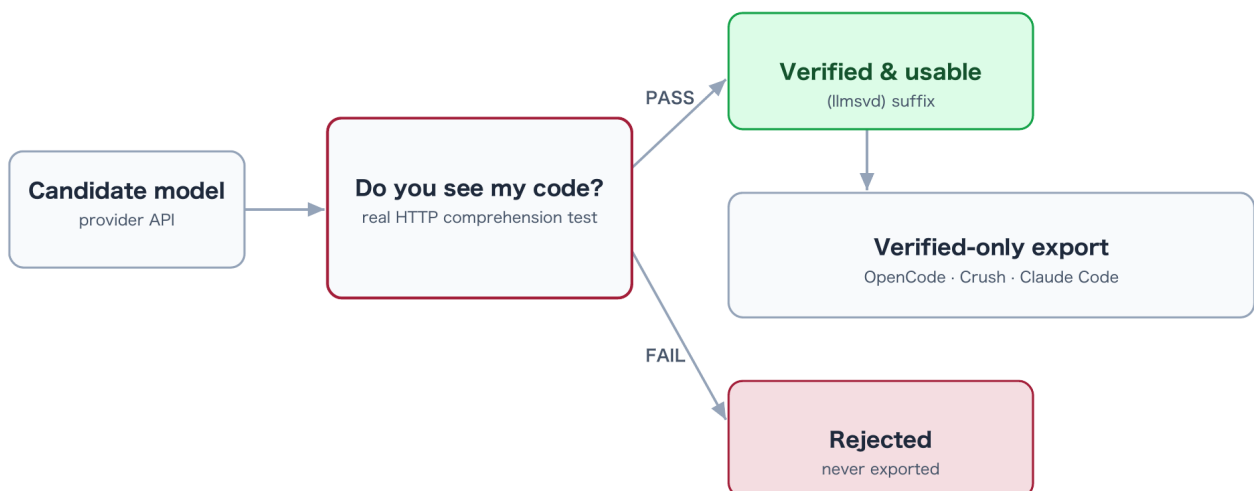
heterogeneous
nodes T1-T8

L0 · Hardware Substrate (GPU → edge SBC)

Node tiers T1 (datacenter GPU) → T8 (handheld), unified under one control plane

HelixCluster — 一种面向 AI 计算的分布式操作系统，从数据中心 GPU 到边缘手持设备，均由单一控制平面统一管理。

LLMsVerifier — mandatory verification gate



LLMsVerifier — 验证、监控、优化：LLM/供应商/验证元数据的唯一真实来源，以强制性模型理解检查为准入门槛。