VASIC DIGITAL · EVIDENCE-BASED PORTFOLIO

Miloš

Portfolio · Helix family · vasic-d

A unified, evidence-based portfolio — the Helix product family, the vasic-digital utility fleet, and the Server .Factory toolchain, on a shared engineering Constitution

+140	43	25	439
REPOSITORY FLEET	LLM PROVIDERS	LINUX DISTROS PROVISIONED	PASSING TESTS · MAIL SERVER FACTORY

١. نظرة عامة — الصورة الكاملة أولاً

هذه محفظة موحدة واحدة يستخدمها كلّ من vasic.digital وmilosvasic.ru. وهي لا تصف مجموعة متفرقة من المشاريع الجانبية، بل **أسطولاً** مصمماً بعناية: **تطبيقات منتجات كبيرة** مبنية على **عشرات الوحدات الصغيرة المستقلة المختبرة بشكل منفصل**، حيث يحكم الكل إطار هندسي مشترك **Constitution**، ويتحقق منه نظام **ضمان الجودة المضاد للادعاء**. هذا الهيكل هو ما يميزها. فمعظم المحافظ هي مجرد قائمة بالأشياء التي بُنيت؛ أما هذه فهي نظام يُجمع فيه كل منتج من أجزاء مجربة وقابلة لإعادة الاستخدام، وكل جزء يخضع لنفس القواعد غير القابلة للتفاوض، وكل قدرة معلن عنها مدعومة بأدلة ملموسة. اللغة السائدة هنا هي **Go**، مع استخدام Kotlin/KMP وTypeScript/React وPython وSwift وShell حيثما تناسب كل منها بشكل حقيقي — Go للخدمات والمكتبات عالية الإنتاجية، Kotlin للتزويد والتطبيقات متعددة المنصات، TypeScript للواجهات الأمامية المكتوبة بأنواع البيانات، Python للصق بين AI وML.

ما يجعل هذا العمل متماسكاً هو أن الانضباط هنا ميكانيكي وليس طموحاً. فالإطار المشترك **Constitution** يُنشر كوحدة فرعية في Git ويُورث عبر أسطول يضم أكثر من ١٤٠ مستودعاً، بحيث تنتشر أي تغيير في القاعدة في كل مكان؛ كما يرفض نظام ضمان الجودة المضاد للادعاء تسجيل نجاح أي اختبار دون دليل تشغيلي. عائلة Helix والأسطول الخدمي وأدوات Server Factory هي ثلاث تجليات لفكرة واحدة: ابنِ مرة واحدة، أعد الاستخدام في كل مكان، وأثبت أنه يعمل قبل أن تعتبره مكتملاً.

كل ما يلي يعرض بحسب الأولوية:

١. **أركان الحوكمة وضمان الجودة** — HelixConstitution وHelixQA (الانضباط الذي يجعل الباقي موثوقاً). ٢. **عائلة منتجات Helix** — دورة حياة تطوير AI (HelixTrack أولاً). ٣. **البنية التحتية LLM** — تجريد المزودين والتنسيق والتحقق. ٤. **أدوات vasic-digital المستقلة** — أدوات مستقلة على مستوى المنتجات. ٥. **Server Factory** — إرث أتمتة البنية التحتية (مرتبة أخيراً).

الأطروحة الموحدة: **لا يُعتبر الميزة مكتملة إلا عندما يتمكن مستخدم حقيقي من استخدامها، وهناك دليل ملموس يثبت ذلك.**

---

٢. أركان الحوكمة وضمان الجودة

يأتي هذان العنصران أولاً لأن كل ما سواهما في هذه المحفظة يستمد مصداقيته منهما. فهما معاً يحولان عبارة "ثق بي، إنه يعمل" إلى حقيقة قابلة للتدقيق — حيث يُرمّز الإطار **Constitution** القواعد، بينما يثبت نظام **HelixQA** أنها اتُبعت.

- **HelixConstitution** — كتاب قواعد هندسية شامل وغير مرتبط بمشروع محدد، يُنشر كوحدة فرعية في Git ويُورث عبر أسطول يضم أكثر من ١٤٠ مستودعاً. بوابات أدلة مضادة للادعاء، مناعة ضد الإيجابيات الكاذبة، سلامة البيانات والمضيف، انضباط التغطية؛ وراثة قابلة للتوسيع دون إضعاف؛ بوابات انتشار تبحث عن البنود المطلوبة عبر الأسطول بأكمله؛ وكل بوابة مقترنة باختبار تحور يثبت أنها ليست مجرد خدعة.
- **HelixQA** — تنسيق ضمان الجودة المضاد للادعاء (Go). بنوك اختبارات مكتوبة بلغة YAML بالإضافة إلى جلسات ضمان جودة مستقلة بالكامل تعتمد على LLM والرؤية الحاسوبية عبر أنظمة Android وAndroid TV والويب وسطح المكتب؛ لا يُسجل نجاح أي اختبار دون أدلة ملموسة (لقطات شاشة، سجلات logcat، مقاطع فيديو، تتبع المكدس). نوع الاختبار الإلزامي لضمان الجودة وفقاً للإطار **Constitution** (§١١.٤.١٦٩).

٣. عائلة منتجات Helix

تعدّ سلسلة Helix الرائدة عائلة متكاملة من المنتجات تغطي دورة حياة تطوير AI بأكملها، بدءاً من التخطيط والمواصفات مروراً بالبناء والذاكرة والترجمة وصولاً إلى التسليم. كل منتج منها قائم بذاته، إلا أن القصد من التصميم هو تكاملها معاً: نفس الحوكمة، نفس الوحدات القابلة لإعادة الاستخدام، ونفس الانضباط القائم على الأدلة تحت مظلة واحدة.

- **HelixTrack** — بديل JIRA للعالم الحر؛ الرائد في خط منتجات Helix-Track.
- **HelixAgent** — خدمة LLM التجميعية: تتناقش نماذج متعددة وتصدر الإجابة التي تتفق عليها، مع اختيار المزودين بناءً على التحقق.
- **HelixCode** — منصة تطوير AI موزعة على مستوى المؤسسات؛ تقسم المهام بين العاملين المدارين بواسطة SSH مع نقاط تفتيش واستعادة تلقائية؛ واجهات REST وCLI وTUI وMCP.
- **HelixCluster** — نظام تشغيل موزع لحوسبة AI، من وحدات معالجة الرسوميات في مراكز البيانات إلى الأجهزة المحمولة الطرفية، تحت لوحة تحكم واحدة.
- **HelixBuilder** — خط إنتاج مدعوم بـAI لبناء التطبيقات، فئة تلو الأخرى.
- **HelixSkills** — نظام مهارات محكوم ومستند إلى دستور خاص بوكلاء CLI من AI (المهارات، خوادم الأدوات MCP، إضافات Claude Code).
- **HelixSpecifier** — التطوير القائم على المواصفات الذي يضبط طقوسه وفقاً لحجم العمل.
- **HelixMemory** — دماغ ذاكرة واحد لوكلاء AI، يجمع بين أربعة محركات رائدة في فئتها.
- **HelixTranslate** — ترجمة كتب تعتمد على النماذج المُتحقق منها؛ مصممة لتكون شفافة، ولا تلجأ أبداً إلى التراجع الصامت.
- **HelixTerminator** — منصة طرفية بلا ثقة: كل جلسة SSH مؤمنة ومشتركة ومساعَدة بواسطة AI.
- **HelixGitpx** — Git موحد عبر عشرات المضيفين؛ مصدر واحد للحقيقة، متكرر في كل مكان.
- **HelixOTA** — تحديثات عبر الهواء عالمية ومستقلة؛ مصممة لتكون بلا أعطال.
- **HelixPlay** — حوّل أي جهاز GPU إلى جهاز سحابي للألعاب خاص بك.
- **Helix-Flow** — منتج استدلال لمنصة Helix. غير مُتحقق منه / معلق بانتظار المصدر: مستودعه العام يحتوي حالياً على ملف README بسطر واحد فقط؛ سيتم تقديمه بالعمق المنتج عند توفر الوثائق الحقيقية.

٤. البنية التحتية لـLLM

تحت المنتجات تكمن الطبقة الأساسية التي تجعلها مستقلة عن المزودين وموثوقة: واجهة واحدة فوق عشرات مزودي LLM، ولوحة تحكم واحدة لوكلاء الترميز غير المرئيين، ومصدر واحد للحقيقة القائم على التحقق. هذه هي الطبقة التي تسمح لكل ما فوقها بتبديل النماذج، والبقاء على قيد العمل أثناء انقطاعات المزودين، ورفض الثقة بأي LLM لا يستطيع إثبات فهمه للمهمة.

- **HelixLLM** — ثنائي واحد بستة أوضاع: استدلال متوافق مع OpenAI وAnthropic عبر llama.cpp، HTTP/3، محلي، سلسلة تراجع مُقيّمة، خط أنابيب RAG، ووكلاء ReAct.
- **LLMProvider** — واجهة واحدة لـ٤٣ مزوداً، مع قواطع دوائر وإعادة محاولة وآليات صحية مدمجة.
- **LLMOrchestrator** — لوحة تحكم واحدة لكل وكلاء الترميز غير المرئيين من CLI (OpenCode، Claude Code، Gemini، Junie، Qwen Code).
- **LLMsVerifier** — تحقق وراقب وحسّن: المصدر الوحيد للحقيقة الخاصة ببيانات LLM والمزودين والتحقق، مع بوابة إلزامية لفهم النموذج.

المحتوى

٥. أدوات فاسيك الرقمية

أدوات مستقلة ذات مستوى إنتاجي، كل منها يحل مشكلة معقدة بذاتها — وهي، ليس من قبيل الصدفة، تختبر أسطول الوحدات القابلة لإعادة الاستخدام بكثافة. تتراوح بين نظام وسائط متعدد البروتوكولات مرن إلى خط إنتاج دورات تعليمية من ماركداون إلى فيديو، وصولاً إلى محرك مزامنة الوثائق/قواعد البيانات المحتوى-المهشّم؛ بعضها صريح بشأن مستوى نضجه، وهذه العلامات تُحفظ ولا تُدفن.

- **Catalogizer** — إدارة مجموعات الوسائط متعددة البروتوكولات (SMB/FTP/NFS/WebDAV/محلي)، مشفرة وقابلة للاستضافة الذاتية؛ واجهة Go/Gin مع React + API؛ مراقبة مرنة تتحمل الانقطاع؛ مبنية على ٢١ وحدة فرعية من `*.digital.vasic`.
- **Courses-Creator** — خط إنتاج دورات تعليمية من ماركداون إلى فيديو؛ إثراء متعدد (Bark TTS، LLM/SpeechT5)، ومشغلات سطح المكتب/الهواتف/الويب؛ وضع تشغيل سلس بدون مفتاح API.
- **VisionEngine** — مجموعة أدوات Go المفككة التي تدمج الرؤية الحاسوبية الكلاسيكية مع رؤية LLM متعددة المزودين؛ رسوم بيانية للملاحة مع BFS وتصدير إلى DOT/JSON/Mermaid؛ بوابة بناء OpenCV محددة بالعلامات.
- **DocProcessor** — خريطة الميزات من الوثائق مع تتبع تغطية التحقق؛ استخراج LLM أو استخراج قائم على الاستدلال/غير متصل؛ ترخيص Apache-2.0.
- **Docs Chain** — مزامنة وثائق/قواعد بيانات ثنائية الاتجاه ذرية المحتوى-المهشّم (إعادة حساب تدريجية على نمط Salsa عبر DAG). المراحل ١–٥ مكتملة؛ المراحل ٦–٧ مخطط لها.
- **Herald** — إشعارات متعددة القنوات موثوقة مع تحليل نية ثلاثي المستويات بلغة طبيعية (أمر → LLM → توضيح)؛ أول مستهلك لـ Docs Chain.
- **task_bridge** — مزامنة مهام/لوحات ثنائية الاتجاه مفككة (SQLite مصدر وحيد للحقيقة ⍰ الوثائق ⍰ ClickUp). هيكل المرحلة الأولى — منطق المزامنة لم يُنفذ بعد.
- **Vasic Digital مجموعة الوحدات القابلة لإعادة الاستخدام** — "المكتبة القياسية" لـ `*.digital.vasic`: أساسيات البنية التحتية، لبنات بناء AI، وحاجز دفاعي لـ LLM، بالإضافة إلى مرآة Kotlin Multiplatform. عدة مستودعات مؤسسية هي في مرحلة السقالات/قيد التطوير — مُعلّمة، وغير منشورة.

٦. Server Factory (إرث أتمتة البنية التحتية — مُصنّف أخيراً)

مُصنّف أخيراً عن قصد، وليس لجودة الأدوات: سلسلة أدوات Server Factory سبقت خط AI وتُظهر أين تشكلت فلسفة "ابنِ مرةً، أعد الاستخدام في كل مكان" لأول مرة. منتجها الرائد — خادم بريد تُصفه في JSON وتُنشره في أي مكان — هو منتج ناضج ومُختبر جيداً؛ أما بقية الأدوات المصاحبة فتُقدّم بمستويات نضجها الحقيقية دون تجميل.

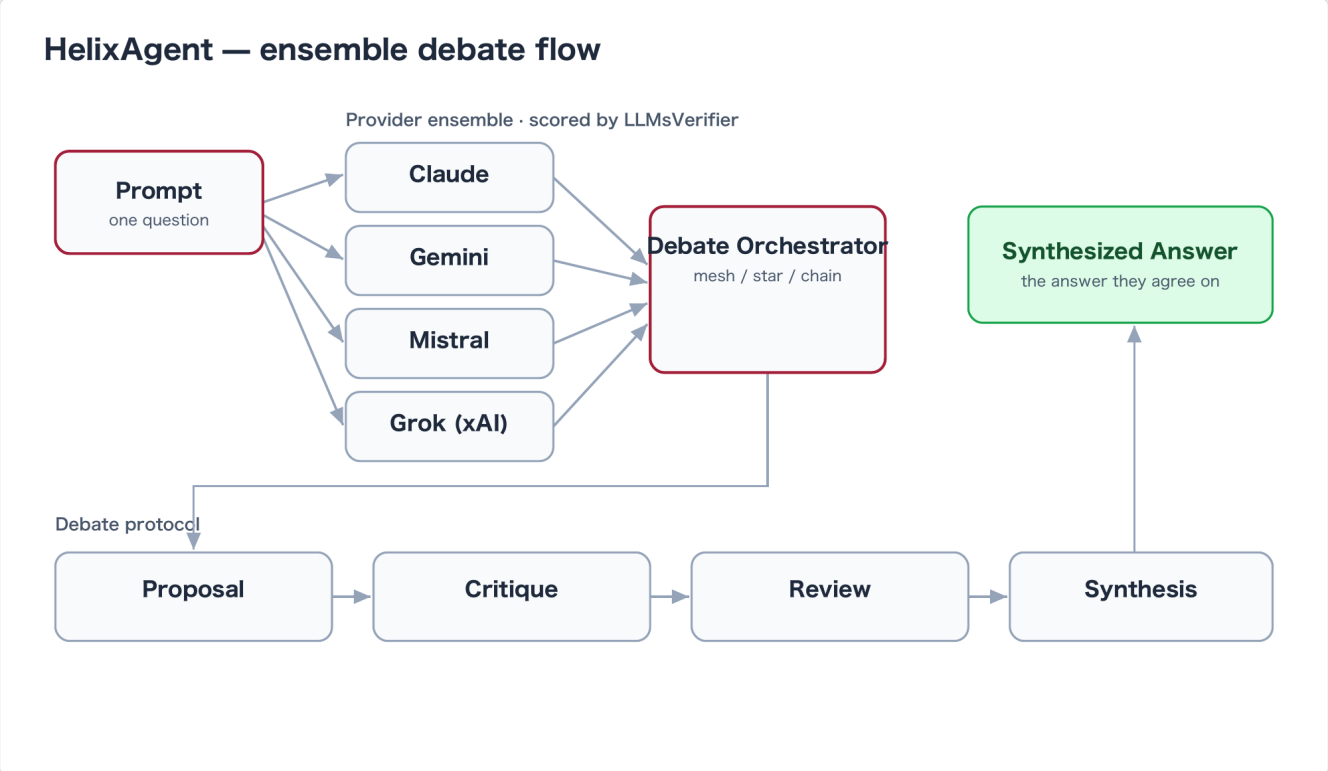
- **Mail Server Factory** — خادم بريد Docker مُعد بالكامل عبر JSON التقريرية عبر ١٢ نوع اتصال و٢٥ توزيعة لينكس؛ أمان مؤسسي؛ يُبلغ عن اجتياز ٤٣٩ اختباراً وبوابة SonarQube نظيفة. المنتج الرائد لمنظمة Server-Factory.
- **Server Factory الإطار الأساسي** — محرك Kotlin المشترك الذي تبنى عليه كل المصانع.
- **Qemu-Utils** — صور آلات افتراضية QEMU تُدار كقطع أثرية: تحميل/تخزين مؤقت/تشغيل، ضغط/نشر، شبكات جسرية/TAP، تثبيت عبر ISO؛ لينكس وماك.
- **Parallels-Utils** — ضغط صور الآلات الافتراضية Parallels (ماك)، نشر واسترجاع عبر ملفات إعدادات بسيطة.
- **Server Factory — المكونات الإضافية** — مصانع خدمات (ويب/SonarQube/وكيل التخزين المؤقت)، حزم التعريفات، وأدوات مساعدة. مصانع الخدمات موثقة بشكل مؤقت — غير مُتحقق منها/في مراحل مبكرة.

المحتوى

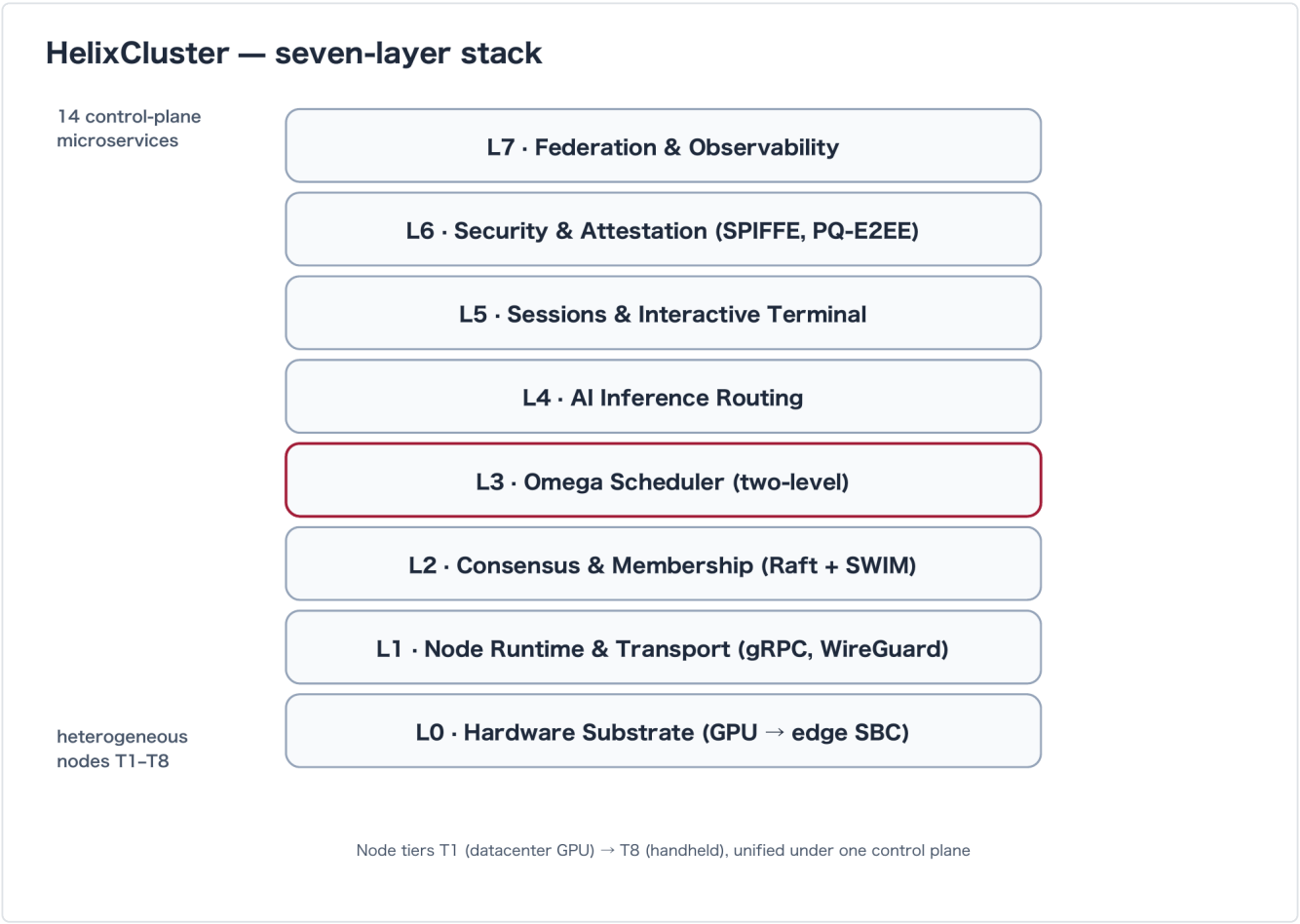
٧. مؤشر التكنولوجيا (قائم على الأدلة)

- **اللغات:** Go (السائدة)، Kotlin وShell، Kotlin Multiplatform، TypeScript، Python، Swift؛ PL/pgSQL؛ TLA+ (المواصفات الرسمية في helix_cluster).
- **AI / LLM:** وصول إلى أكثر من ٤٠ مزودًا، MCP، RAG، قواعد البيانات vector وembeddings، التخطيط (HiPlan/MCTS/شجرة الأفكار)، LLMOps، المقارنة المعيارية (SWE-bench/HumanEval/MMLU)، TTS (Bark/SpeechT5)، الرؤية الحاسوبية + LLM-vision، الضوابط/الفريق الأحمر.
- **الواجهة الخلفية:** Kafka /Gin، gRPC + Protobuf، HTTP/3 (QUIC)، WebSockets، Angular، React، RabbitMQ.
- **البيانات:** PostgreSQL، SQLite، SQLCipher، Redis، Neo4j، ClickHouse، MinIO/S3/GCS/Azure.
- **البنية التحتية/DevOps:** Docker وCompose، Kubernetes + Helm، Prometheus + Grafana، OpenTelemetry؛ QEMU/Libvirt/Parallels، GitHub Actions، Gradle، Make.
- **الاختبار/ضمان الجودة:** HelixQA، أطر التحدي مع بوابات التحور، `go test -race`، أدوات اختبار الانحدار البصري، اختبار الأجهزة عبر ADB، SonarQube، فحص الأمان (semgrep/gosec/trivy/snyk/gitleaks/nancy)، التحقق من النماذج عبر TLA+.

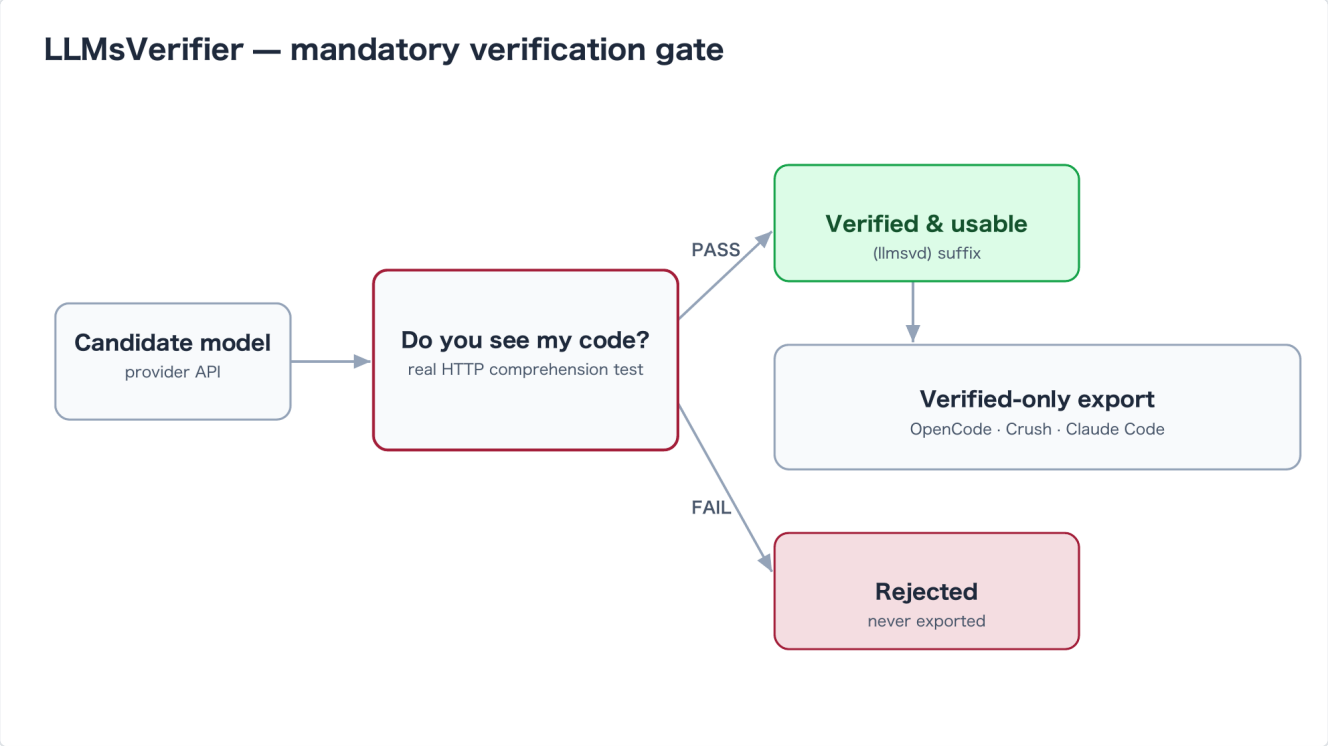
Selected architecture



an ensemble LLM service where providers debate under a four-stage protocol and ship the answer they — **HelixAgent** .agree on, scored by LLMsVerifier



> a distributed operating system for AI compute, from datacenter GPUs to edge handhelds, under one — **HelixCluster** .control plane



> verify, monitor, optimize: the single source of truth for LLM/provider/verification metadata, gated by a — **LLMsVerifier** .mandatory model-comprehension check