

Сравнение CodeAlive и 1C AI Stack

Ось	CodeAlive	1C AI Stack
Основная цель	Глубокий контекст кодовой базы для AI-агентов, умный code review и анализ сложных проектов	Платформа для 1C-ландшафта: DevOps, BA→Dev→QA, DR, observability, security, GitOps, FinOps, LLM-интеграции
Главный продукт	Agent-based Code Review + multi-repo code intelligence	Scenario Hub (BA→Dev→QA, DR, security-audit) + Orchestrator + ToolRegistry; набор сервисов/скриптов вокруг 1C и инфраструктуры
Модель данных	Единый граф знаний по коду (multi-repo, связи между модулями и сервисами)	Несколько уровней: граф по 1C/конфигурациям (Neo4j), BA-данные, метрики, DR/инфра-артефакты, но нет единого универсального графа по всем языкам/репо
Роль MCP/IDE	MCP/IDE — основной интерфейс к «живому» коду с полным контекстом»	MCP — один из фронтов (IDE), основное выполнение вынесено в HTTP/CLI/playbooks через Scenario Hub и Orchestrator
Сценарии разработчика (Dev)	Code review, анализ архитектуры, подсказки по изменениям, интеграция с Jira	Анализ 1C-конфигураций, генерация кода/тестов, SQL-оптимизация, BA→Dev→QA e2e, частично code review через Agent Framework
Сценарии Ops/SRE	Фокус минимальный, опосредован через качество кода и PR-процесс	DR rehearsal, постмортемы, GitOps/ArgoCD, Linkerd/Istio, Vault/CSI, SLO/alerts, DORA, FinOps, preflight/policy-check, security-audit
Сценарии ВА/менеджмента	Через задачи/тикеты (например, Jira), продукт Dev-центристский	Отдельный ВА-агент, requirements extraction, BA roadmap, сценарии BA→Dev→QA, интеграции Jira/Confluence/Docflow, статус/weekly reports
Работа с несколькими репозиториями	Multi-repo — одна из ключевых фич: единый контекст по нескольким репо	Есть GitOps/infra-репо и код, но нет единого CodeAlive-подобного графа по всем репозиториям и языкам
Архитектурный центр	Knowledge graph + AI-агенты для кода	Scenario Hub + Orchestrator + ToolRegistry; вокруг — LLM gateway, security-полиси, DR, GitOps, observability
Безопасность	Фокус в основном через качество кода и практики review	Полноценный security-контур: secret scanning, Rego/Conftest (K8s/Terraform/GitOps), Semgrep, DR/security-схемарии, policy waivers, runtime checks
Observability/метрики	Главным образом в контексте качества кода и процессов разработки	Prometheus/Grafana/Alertmanager для AI/инфры, DORA, FinOps, AI-perf (Orchestrator/Kimi/Qwen), Telegram alerts, runbooks
Сильные стороны	Сильнее в едином живом контексте кода и agent-based code review	Сильнее в платформенности: прод-процессы, DR, GitOps, security, BA/Dev/QA-схемарии, 1C-специфика и операционный контур
относительно другого		

Пространство
роста
относительно
другого

Требует более полного
DevOps/Prod-контекста

Не хватает «CodeAlive-подобного»
универсального кодового графа и
фронтового продукта
agent-code-review