

Ruflo (Claude-Flow): рой из 60+ агентов вокруг Claude Code

Claude Code сам по себе работает в одиночку — один агент, одна сессия, память только на время чата. Ruflo (бывший Claude-Flow) добавляет вокруг него слой координации: десятки специализированных агентов, общую память между сессиями и автоматическое распределение задач. Разбор того, что это, какие цифры подтверждаются и как это ставится.

1. Что такое Ruflo (бывший Claude-Flow)

- Проект Claude-Flow переименован в Ruflo — это тот же репозиторий и то же сообщество под новым именем.
- По сути это агентский мета-харнесс для Claude Code и Codex: не заменяет модель, а даёт готовому агенту инструменты, память, циклы выполнения и контроль, чтобы задача доводилась до конца, а не обрывалась на середине.
- Открытый код на MIT, около 62,8 тысяч звёзд на GitHub и больше 1500 релизов — активно развивающийся проект.

2. Ключевые возможности

- **Агенты:** свыше 60 специализированных ролей — кодер, тестировщик, ревьюер, архитектор, специалист по безопасности и другие, которые самоорганизуются в рой (swarms).
- **Queen-координация:** иерархия с ведущим агентом («queen»), консенсус через Raft, Byzantine или Gossip-протоколы — вместо ручной оркестрации каждого шага вручную.
- **Hive-mind память:** векторная память на HNSW-индексе (AgentDB) с почти мгновенным поиском — контекст и решения сохраняются между сессиями, а не теряются с закрытием чата.
- **Самообучение:** механизмы SONA и ReasoningBank учат агентов на прошлых успешных траекториях — со временем система точнее выбирает подход к похожим задачам.
- **Роутинг:** задачи автоматически распределяются между 5 провайдерами моделей (Claude, GPT, Gemini, Cohere, Ollama) с отказоустойчивостью, если один из них недоступен.

3. Реальные цифры

- **Точность маршрутизации:** около 89% — насколько точно система сама распределяет задачи между агентами.
- **Экономия и решаемость:** порядка 75% экономии стоимости и 84.8% solve-rate на тестовых задачах — цифры ориентировочные и сильно зависят от конкретной задачи и модели, а не гарантированный результат на любом проекте.
- **Сравнение с другими фреймворками:** по опубликованным бенчмаркам Ruflo обгоняет LangGraph, AutoGen и CrewAI по холодному старту, одиночному запросу и потреблению памяти в 1.3–1953 раза в зависимости от сценария; методология и сырые данные — в открытом виде у разработчика.

4. Установка

- **Быстрый старт (все платформы):** `npx ruflo@latest init wizard` — интерактивный визард, работает одинаково на macOS, Linux и в нативных Windows PowerShell/cmd.
- **Одной командой (POSIX):** `curl -fsSL https://cdn.jsdelivr.net/gh/ruvnet/ruflo@main/scripts/install.sh | bash` — для macOS, Linux, WSL и Git-Bash.
- **Глобально через npm:** `npm install -g ruflo@latest`.
- **MCP-сервер в Claude Code:** `claude mcp add ruflo -- npx ruflo@latest mcp start`.

- **Облегчённый вариант:** через маркетплейс плагинов Claude Code — `/plugin marketplace add ruvnet/ruflo`, затем `/plugin install ruflo-core@ruflo` и нужные плагины; даёт только слэш-команды без полноценного MCP-сервера.

5. Что важно учитывать

- **Разница путей установки:** плагин из маркетплейса — облегчённая версия без MCP-сервера (`memory_store`, `swarm_init` и другие инструменты недоступны); за полным функционалом — именно CLI-установщик (`npx ruflo init`).
- **Безопасность:** один из недавних релизов — исправление уязвимости в MCP-мосту, стоит обновляться до актуальной версии, а не оставаться на старой.
- **Federation:** функция федерации агентов между разными машинами передаёт данные только после снятия персональных данных и шифрования, но это ещё относительно новая часть системы — стоит проверять актуальный статус перед использованием на продакшене.