

Vibe Coding

Prakticky

nástroje 2026

Kompletní praktický průvodce: Claude Code · Gemini CLI · Codex CLI · OpenCode · Antigravity · GitHub Copilot · Continue.dev · Cursor · Claude Cowork. Aktuální verze, instalace, srovnání terminologie a netriviaální rozdíly.

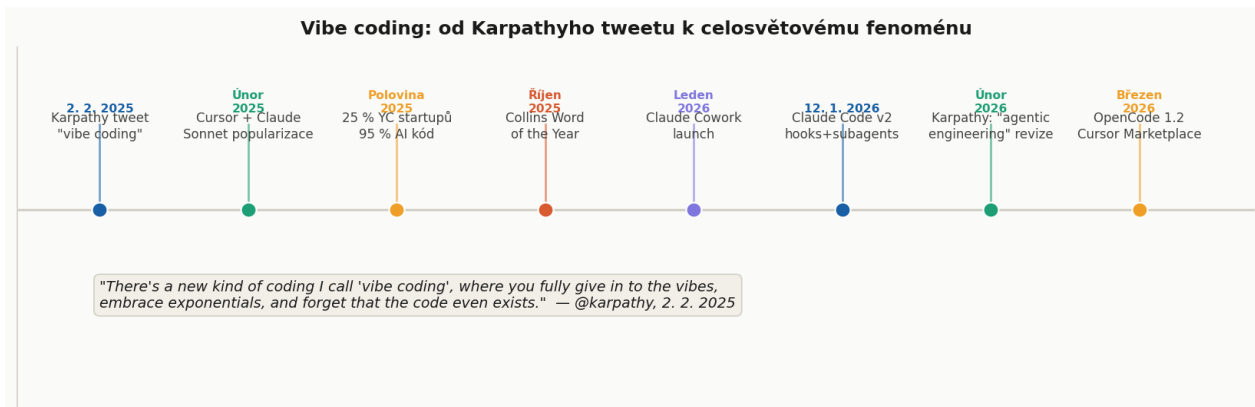
9 nástrojů · terminologická mapa · srovnání cen · Karpathyův citace · březen 2026

„There’s a new kind of coding I call ‘vibe coding’, where you fully give in to the vibes, embrace exponentials, and forget that the code even exists. It’s possible because the LLMs are getting too good. Also I just talk to Composer with SuperWhisper so I barely even touch the keyboard.“

— Andrej Karpathy, 2. února 2025 (X / Twitter, status/1886192184808149383)

Co je vibe coding a proč ho používat

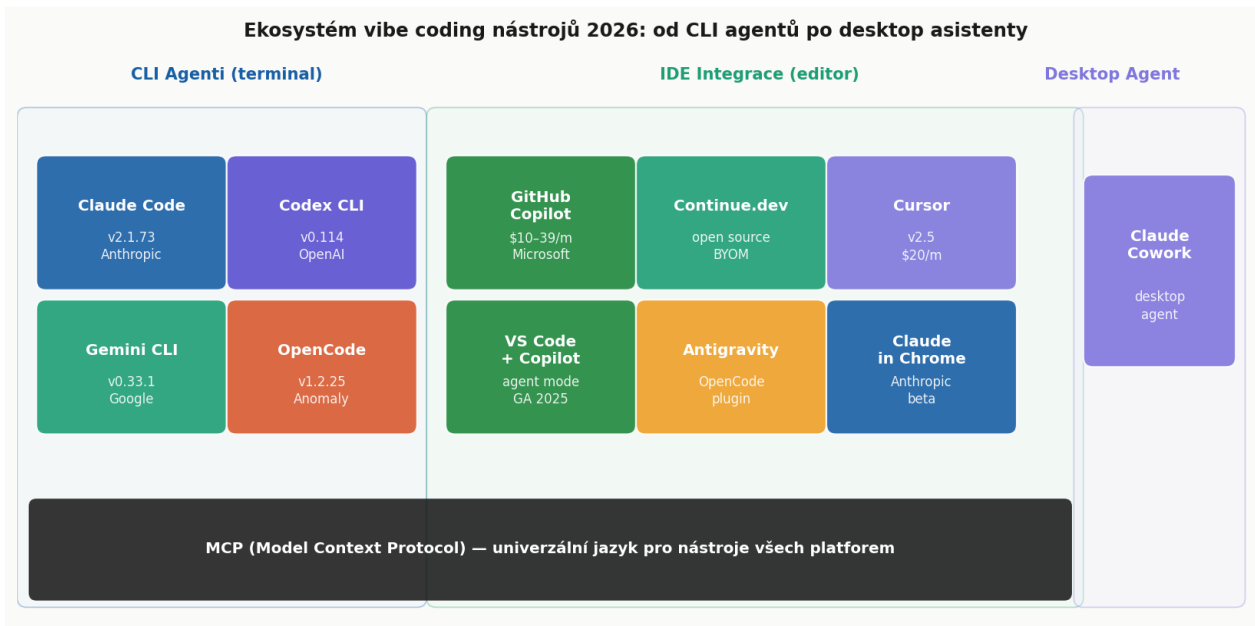
Vibe coding = přístup k práci s počítačem, kde AI agent provádí implementační detaily a ty řídíš směr. Ne nutně kód — stejně dobře pro poznámky, analýzu dat, dokumenty, správu souborů nebo e-maily.



Obr. 1 — Od tweetu k "Word of the Year 2025": vibe coding se během roku přeměnil z Karpathyho nápadu na mainstreamový workflow. 25 % YC W25 startupů: 95 % AI kód.

Proč používat vibe coding (i pro neprogramatory)

Rychlejší iterace: od nápadu k funkčnímu výsledku bez zdržení syntaxí. Analýza dat bez Excelu. Dokumenty bez Wordu. Automatizace bez scriptu. Karpathy (2026): dnes rozlišuje "vibe coding" (throwaway) vs "agentic engineering" (produkční, s dohledem).



Obr. 2 — Ekosystém nástrojů 2026: CLI agenti (terminal), IDE integrace (editor), desktop agent. MCP (Model Context Protocol) je univerzální standard pro rozšíření — funguje všude.

Terminologická mapa: co kdo jak nazývá

Největší zmatek v ekosystému je, že každý nástroj vymyslel vlastní slovník pro ekvivalentní koncepty. Claude Code říká "MCP servers + hooks", Gemini CLI "Extensions", Codex CLI "MCP + Skills + Apps", OpenCode "Plugins", Cursor "Plugins s Marketplace". Všechny podporují MCP — jediný opravdu přenosný formát.

Terminologická mapa: co každý nástroj jak nazývá (verze březen 2026)

| | Claude Code | Gemini CLI | Codex CLI | OpenCode | Copilot | Continue.dev | Cursor |
|---------------------|-------------------------|-------------------------|---------------------|----------------------|------------------------------|--------------------|-----------------------|
| Projekt kontext | CLAUDE.md | GEMINI.md | AGENTS.md | AGENTS.md | .github/copilot-instructions | .continuerules | .cursorsrules |
| Ext. systém | MCP servers + Hooks | Extensions | MCP+Skills + Apps | Plugins | Skills+Agents + Extensions | Configs (Hub) | Plugins (Marketplace) |
| Ext. protokol | MCP | MCP | MCP | MCP | MCP | MCP | MCP |
| BYOM | Ne - jen Anthropic | Ne - jen Google | Ne - jen OpenAI | Ano 75+ provideri | cast. - model picker | libovolný provider | cast. - model picker |
| Cross-session paměť | Auto memory + CLAUDE.md | GEMINI.md + save_memory | AGENTS.md | AGENTS.md (no auto) | Agentic Memory 28 dní | Žádná vbuildovaná | Žádná zabudovaná |
| Cena | \$20/m nebo API | ZDARMA 1000req/den | Součást Plus \$20/m | ZDARMA (vlastní API) | \$10/m Pro | ZDARMA (BYOM) | \$20/m Pro |

Obr. 3 — Terminologická mapa: zelené pole u "Ext. protokolu" = MCP je všude stejný. Hlavní rozdíl: BYOM (bring your own model) mají jen OpenCode a Continue.dev.

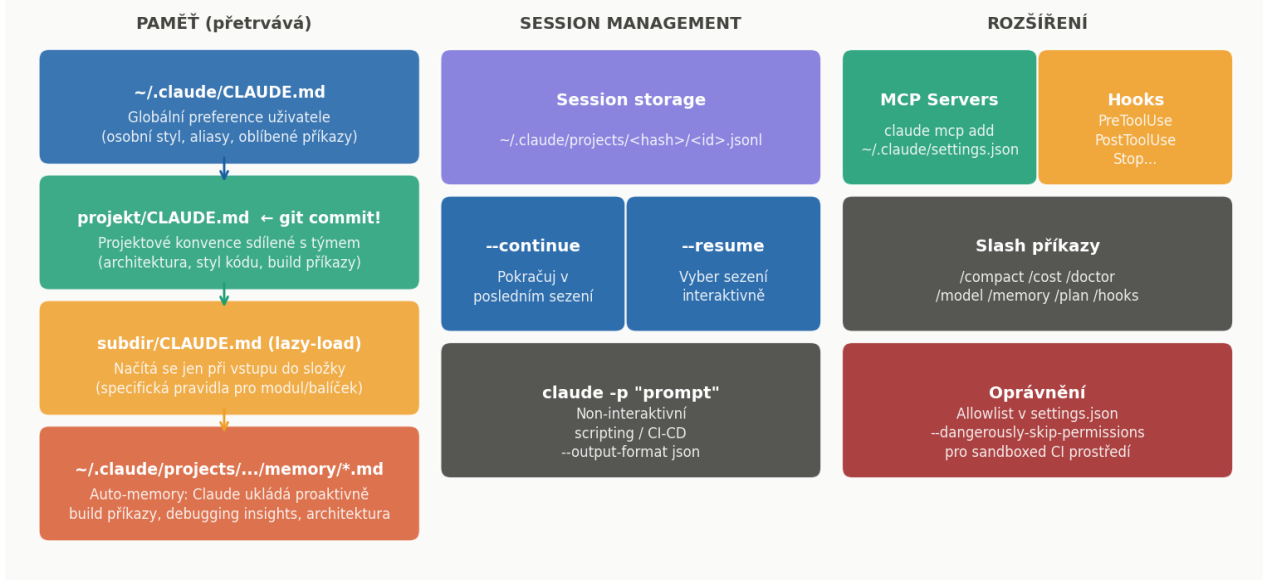
| Koncept | Claude Code | Gemini CLI | Codex CLI | OpenCode | Copilot/Cursor |
|---------------------|-------------------------|-------------------------|-------------------|-----------|---|
| Projekt kontext | CLAUDE.md | GEMINI.md | AGENTS.md | AGENTS.md | .github/copilot-instructions .cursorsrules |
| Ext. ekosystem | MCP servers + Hooks | Extensions | MCP+Skills + Apps | Plugins | Skills+Agents + Extensions / Plugins |
| BYOM model | Ne | Ne | Ne | Ano (75+) | Cast. / Ne |
| Cross-session paměť | Auto memory + CLAUDE.md | GEMINI.md + save_memory | AGENTS.md | AGENTS.md | Cop: 28-dní Cursor: žádná |
| Open source | Ne | Ano (Apache) | Ano (Apache) | Ano (MIT) | Ne / Ne |

Claude Code — Anthropic

Verze: 2.1.73 (březen 2026) · Docs: docs.anthropic.com/claude-code

```
# Instalace (doporučená metoda) curl -fsSL https://claude.ai/install.sh | bash brew
install --cask claude-code # macOS irm https://claude.ai/install.ps1 | iex # Windows #
Starejší metoda (vyžaduje Node 18+) npm install -g @anthropic-ai/claude-code
```

Claude Code: hierarchie paměti a session management



Obr. 4 — Claude Code: 4-úrovňová hierarchie paměti (globální → projekt → subdir → auto-memory), session management (--continue / --resume / -p scripting), MCP servery, hooks a oprávnění.

CLAUDE.md — paměť projektu

```
# Hierarchie (specifictější přepisuje obecnější) ~/.claude/CLAUDE.md # globální preference
uzivatele ./CLAUDE.md # projektové konvence - COMMIT do gitu!
./claude/CLAUDE.md # alternativní umístění ./subdir/CLAUDE.md # lazy-load při vstupu do
složky ~/.claude/projects/*/memory/*.md # auto-memory (Claude ukládá proaktivně)
```

Session management a scripting

```
claude --continue # pokračuj v posledním sezení claude --resume # interaktivní výběr
sezení claude --resume # konkrétní sezení # Non-interaktivní scripting (CI/CD) claude -p
"spust testy a oprav chyby" --output-format json claude -p "analyzuj diff"
--output-format stream-json
```

MCP servery a hooks

```
# Přidat MCP server claude mcp add filesystem -- npx
@modelcontextprotocol/server-filesystem /tmp claude mcp add github -- npx
@modelcontextprotocol/server-github # Hooks (v .claude/settings.json nebo
~/.claude/settings.json) # Typy: PreToolUse, PostToolUse, PostToolUseFailure, Stop,
SessionStart... # Použití: blokování nebezpečných příkazů, auto-formátování, notifikace
```

Klíčové slash příkazy

Nejpoužívanější příkazy

`/compact` — komprimuj kontext · `/cost` — statistiky API · `/doctor` — troubleshooting · `/memory` — správa paměti · `/model` — změna modelu · `/plan` — plánovací režim · `/hooks` — konfigurace hooků · `/mcp` — status MCP serverů · `/voice` — hlasový vstup

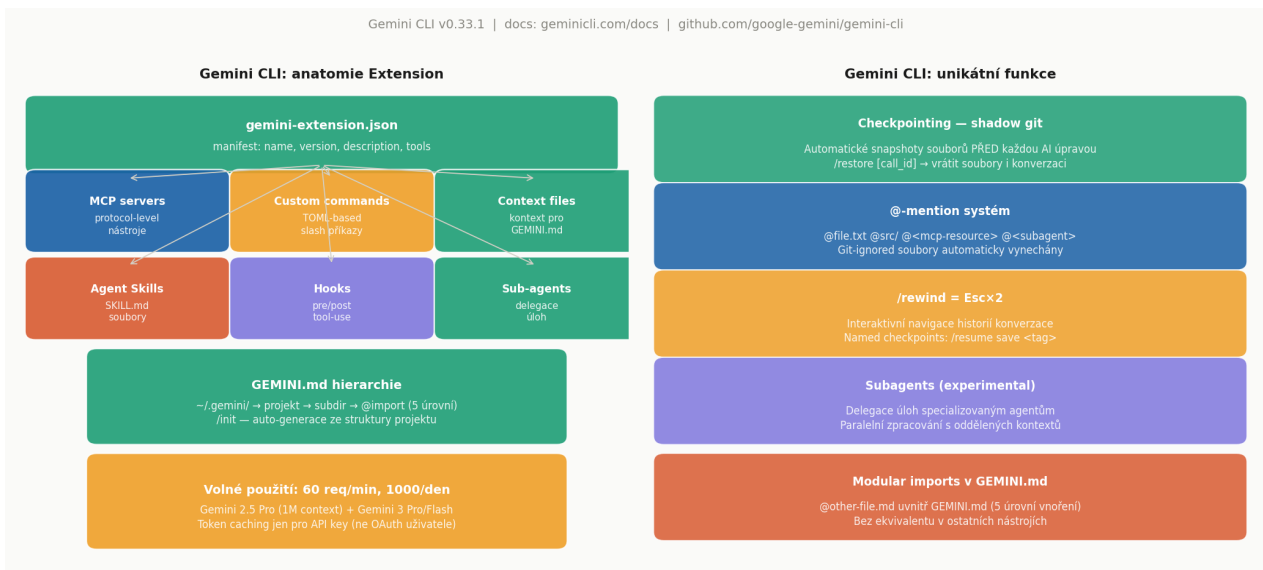
Gemini CLI — Google

Verze: 0.33.1 (březen 2026) · Docs: geminicli.com/docs · github.com/google-gemini/gemini-cli

Jediný skutečně bezplatný nástroj v kategorii

ZDARMA: 60 požadavků/min, 1 000/den s Google účtem. Přístup k Gemini 2.5 Pro (1M token kontext) + Gemini 3 Pro/Flash. Token caching jen pro API key (ne OAuth uživatele).

```
npm install -g @google/gemini-cli npx @google/gemini-cli # bez instalace google gemini #
pokud Google Cloud Shell # Autentizace gemini auth login # OAuth přes browser
(doporučené, zdarma) export GEMINI_API_KEY=... # nebo API klíč (placený)
```



Obr. 5 — Gemini CLI: Extension ekosystém (MCP+commands+hooks+skills v jednom balíčku) a unikátní funkce: checkpointing, @-mention, /rewind a modulární GEMINI.md importy.

GEMINI.md a @-mention systém

```
# GEMINI.md hierarchie (concatenace do system promptu) ~/gemini/GEMINI.md # globální
./GEMINI.md → rodicovske slozky # projekt (az do trusted root) # Modulární importy
uvnitř GEMINI.md (5 úrovní) @./shared/coding-rules.md # @-syntax uvnitř GEMINI.md! #
@-mention v rozhovoru @path/file.txt @src/ @ @ /init # auto-generace GEMINI.md ze
struktury projektu
```

Checkpointing a session management

```
# Checkpointing (výchozi: vypnuto) # Automatické snapshoty PRED každou AI úpravou souboru # Shadow git repo: ~/.gemini/history// /restore [tool_call_id] # vrát soubory + konverzaci /rewind nebo Esc+Esc # interaktivní navigace historií /resume save # pojmenovaný checkpoint # Sessions # Uloženy: ~/.gemini/tmp//chats/ (30-dní retence) # Nastavení: ~/.gemini/settings.json
```

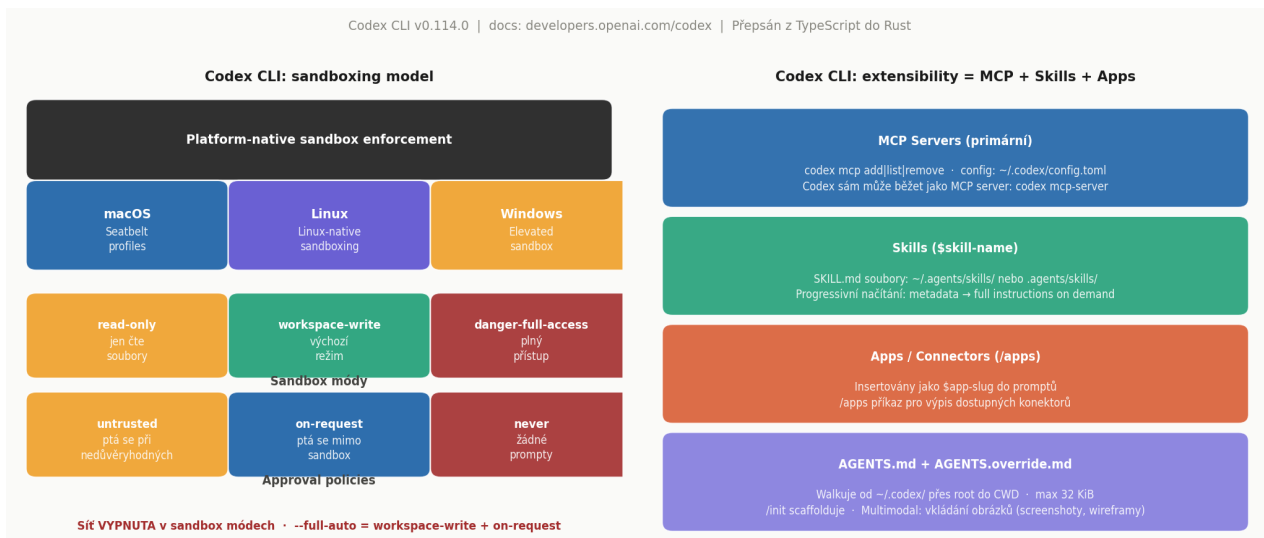
Codex CLI — OpenAI

Verze: 0.114.0 (březen 2026) · Docs: developers.openai.com/codex · Přepsán do Rust

Klíčový diferenciator: platform-native sandboxing

macOS: Seatbelt profily. Linux: Linux-native sandbox. Windows: elevated sandbox. Síť VYPNUTA v sandbox režimech. Nejpro-bezpečnostnější nástroj v ekosystému. Součást ChatGPT Plus (\$20/m) — není nutný zvláštní API klíč.

```
npm i -g @openai/codex brew install --cask codex # macOS # Sandbox módy codex --sandbox read-only # jen čte codex --sandbox workspace-write # výchozi (čte+píše do projektu) codex --full-auto # workspace-write + on-request approval # Approval policies # untrusted = ptá se při nedůvěryhodných příkazech # on-request = ptá se pouze překročí-li sandbox # never = žádné prompty
```



Obr. 6 — Codex CLI: sandbox módy (read-only / workspace-write / danger) kombinovány s approval policies. Extensibility: MCP servery (primární) + Skills (\$skill-name) + Apps / Connectors (/apps).

AGENTS.md a extensibility

```
# AGENTS.md (project context) # Discovery: ~/.codex/ → project root → CWD # AGENTS.override.md má prioritu v libovolném adresáři /init # scaffolduje AGENTS.md # MCP servery codex mcp add filesystem -- npx @modelcontextprotocol/server-filesystem / codex mcp list codex mcp-server # Codex jako MCP server pro jiné nástroje! # Skills ($skill-name v promptu) # Umístění: ~/.agents/skills/ nebo .agents/skills/ # Progresivní načítání: metadata → full SKILL.md on demand
```

OpenCode — SST / Anomaly

Verze: 1.2.25 (březen 2026) · Docs: opencode.ai/docs · github.com/anomalyco/opencode · 120K+ stars MIT

Klíčový diferenciátor: 75+ providerů (BYOM) + TUI

Jediný CLI agent s plným BYOM: Anthropic, OpenAI, Google, Groq, Ollama (lokálně), OpenRouter, AWS Bedrock, Azure, GitHub Copilot, 75+ dalších. TUI s vim keybinds (ctrl+x leader). Zdarma — platíš jen za API.

```
curl -fsSL https://opencode.ai/install | bash npm i -g opencode-ai@latest brew install anomalyco/tap/opencode # Konfigurace: opencode.json (projekt nebo ~/.config/opencode/) # Klíčové příkazy v TUI ctrl+x n # nová session ctrl+x l # seznam sessions ctrl+x m # výběr modelu Tab # přepínání Build (plný prist.)/Plan (readonly) agent
```



Obr. 7 — OpenCode: multi-provider architektura (75+ poskytovatelů), Plugin ekosystém (JS/TS), SQLite session storage, a Antigravity plugin pro bezplatný přístup k Gemini + Claude.

Plugins a MCP

```
# Plugins (opencode.json) { "plugin": ["nazev-pluginu@latest", "./lokalni-plugin"] } # Plugin hooks: tool.execute.before/after, session.created, # session.idle, session.compacted, file.edited, message.updated # MCP server { "mcp": { "github": { "type": "local", "command": "npx @modelcontextprotocol/server-github" } } } # Sdílení session /share → vygeneruje URL: opncd.ai/s/
```

Antigravity — plugin pro bezplatný Gemini + Claude

Repo: github.com/NoeFabris/opencode-antigravity-auth · npm: [opencode-antigravity-auth](https://www.npmjs.com/package/opencode-antigravity-auth)

! Upozornění: Porušuje podmínky použití Google

Antigravity přistupuje k internímu Google IDE API, které není určeno pro veřejné použití. Používej existující Google účty (ne nově vytvořené). Google může odvolat tokeny po bezpečnostních událostech. Plugin odeěrá neplatné účty automaticky.

```
# Instalace (do opencode.json) { "plugin": ["opencode-antigravity-auth@latest"] } #  
Přidat Google účet opencode auth login # opakuj pro více uctu (automatická rotace) #  
Dostupné modely (verze 1.6.0) # Gemini 3 Pro, Gemini 3 Flash # gemini-claude-opus-4-5,  
gemini-claude-sonnet-4-6, gemini-claude-opus-4-6 # Multi-account rotace #  
~/.config/opencode/antigravity-accounts.json # 3 endpoint fallback: daily → autopush →  
prod
```

Jak to funguje technicky

OAuth 2.0 PKCE: browser login → refresh token uložen lokálně. Všechny požadavky jdou přes Google Antigravity endpoints (interní IDE API). Sticky account selection: zůstává na jednom účtu dříve kvóty (Anthropic prompt cache). Při rate-limitu automaticky přepíne na další účet z konfigurace.

IDE integrace: Copilot, Continue.dev, Cursor

| IDE integrace 2026: Copilot Continue.dev Cursor — MCP jako společný standard | | |
|--|---|---|
| GitHub Copilot | Continue.dev | Cursor |
| Agent mode (GA) Autonomní úpravy souborů a terminálové příkazy | BYOM Libovolný provider nebo Ollama lokálně - open source | Composer Multi-file natural language editing - hbitší než Copilot |
| Coding Agent Async: zadej issue → draft PR v cloud VM | @Context providers @file @code @diff @tree @terminal @debugger @repo-map | Codebase indexing 500MB - sub-second queries semantické vyhledávání |
| Agentic Memory 28-dní paměť napříč sesionami a Features | MCP servery Primární ext. mechanismus docs.continue.dev/customize/mcp-tools | Cloud agents (8x) Paralelní agenti v Ubuntu VM max 8 současně |
| Skills .github/skills/SKILL.md Custom Agents: .github/agents/ | Configs (Hub) Sdílení konfigurací přes continue.dev/hub | Plugin system v2.5 Skills + Subagents + MCP + Hooks + Rules v jednom balíčku |
| MCP Servers .vscode/mcp.json Univerzální protokol | Žádná cross-session paměť Kontext jen v rámci sezení Context via rules + MCP | Cursor Marketplace cursor.com/marketplace Amplitude, AWS, Figma, Stripe... |
| Cena Free: 50 req/m Pro 10/m-Business19/u | Cena ZDARMA (BYOM) nebo lokální modely | Cena Pro 20/m-Business40/u vs Copilot Pro \$10/m |

Obr. 8 — Srovnání tří hlavních IDE integrací: GitHub Copilot (agent mode + agentic memory), Continue.dev (open source BYOM, @context providers), Cursor (Composer, Marketplace, cloud agents).

GitHub Copilot — Microsoft

Agent mode (GA 2025): autonomní úpravy souborů a terminálové příkazy. Coding Agent: async — zadej issue, dostaneš draft PR v cloud VM. Agentic Memory (public preview od 15. 1. 2026): 28-denní paměť s citacemi na kód, sdílená mezi features (agent učí → code review použije).

```
# Terminologie: Skills + Custom Agents + Extensions + MCP .github/skills/SKILL.md #  
Agent Skills .github/agents/*.agent.md # Custom Agents s nastaveními .vscode/mcp.json #  
MCP servery # @-mentions v chatu @workspace #file @github # context references # Cena:  
Free (50 req/m) → Pro $10/m → Pro+ $39/m → Business $19/user/m
```

Continue.dev — open source BYOM

Zdarma (Apache 2.0), 31 800+ GitHub stars. Libovolný provider nebo Ollama lokálně. Nepoužívá termín "plugins" — má Configs (sdílené přes Hub), Context Providers (@-systém), Prompts (slash příkazy) a Rules. Není zabudovaná cross-session paměť.

```
# Instalace: VS Code extension nebo JetBrains # Konfigurace: ~/.continue/config.yaml #  
Context Providers (@-mention) @file @code @diff @terminal @tree @problems @debugger  
@repo-map # MCP servery (primární ext. mechanismus) #  
docs.continue.dev/customize/mcp-tools
```

Cursor — VS Code fork s AI-first design

Plugin system (Cursor 2.5, 17. 2. 2026): Skills + Subagents + MCP + Hooks + Rules v jednom balíčku, distribuovány přes Cursor Marketplace. Launch partneři: Amplitude, AWS, Figma, Linear, Stripe, Vercel, Datadog. Cloud agents: až 8 paralelních agentů v Ubuntu VM. Codebase indexing: 500MB, sub-second queries.

Claude Cowork – vibe coding pro neprogramátory

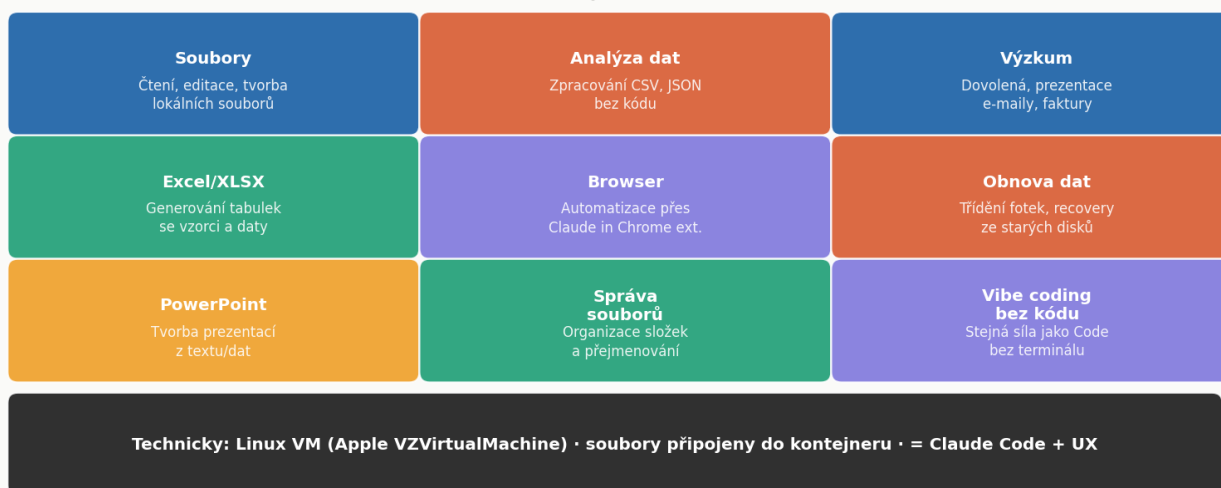
Launch: 12. 1. 2026 · Docs: claude.com/blog/cowork-research-preview · Součást Claude Pro (\$20/m)

Co je Cowork technicky

Záložka v Claude Desktop vedle Chat a Code. Technicky = Claude Code + UX pro neprogramátory. Běží v Linux VM (Apple VZVirtualMachine na macOS) — soubory připojené do kontejneru. Karpathy (parafráze): "regular Claude Code wrapped in a less intimidating default interface."

Claude Cowork: agentic desktop pro neprogramátory (launch 12. 1. 2026)

Claude Desktop → záložka Cowork

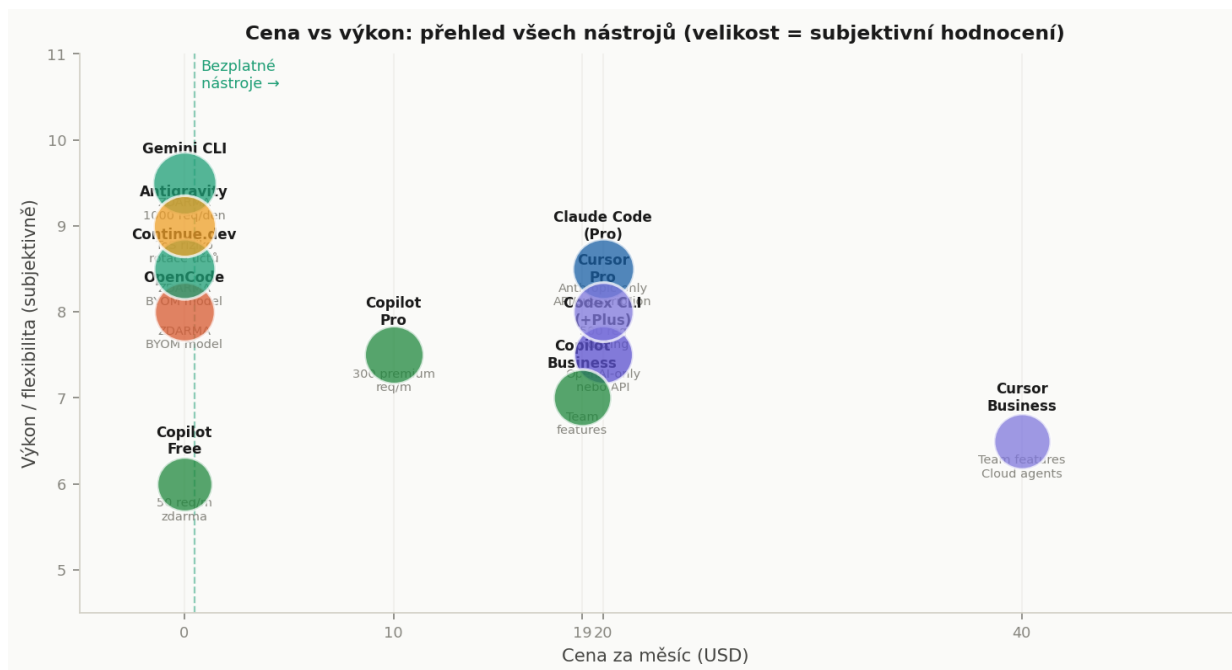


Obr. 9 — Claude Cowork: 9 kategorií use cases od správy souborů po analýzu dat a obnovu fotografií. Dostupný na všech placených plánech bez příplatku.

Typické použití (origin Anthropic: co lidé vyvojáři dělali v Claude Code mimo kód)

Plánování dovolené · Tvorba prezentací · Správa e-mailů · Rušení předplatných · Obnova svátečních fotek z HDD · Excel tabulky se vzorci · Analýza CSV dat · Organizace složek · PDF → Word konverze · Automatizace prohlížeče (přes Claude in Chrome ext.)

Srovnání cen a free tierů



Obr. 10 — Cena vs flexibilita. Gemini CLI je jediný model-lockovaný nástroj s opravdu bezplatným účtem. OpenCode a Continue.dev jsou zdarma s vlastním klíčem. Antigravity: riziko ToS.

| Nástroj | Bezplatný tier | Placený | BYOM | Open source |
|---------------|-------------------|-----------------------------|----------------------|------------------|
| Claude Code | Ne | \$20/m (Pro) nebo API | Ne (jen Anthropic) | Ne |
| Gemini CLI | ANO: 1000 req/den | Google AI Pro | Ne (jen Google) | Ano (Apache 2.0) |
| Codex CLI | Ne | Součást Plus \$20/m | Ne (jen OpenAI) | Ano (Apache 2.0) |
| OpenCode | ANO (BYOM) | Zdarma + API náklady | ANO (75+ provid.) | Ano (MIT) |
| Copilot Free | ANO: 50 req/m | Pro \$10/m, Pro+ \$39/m | Ne | Ne |
| Continue.dev | ANO (BYOM) | Zdarma + API náklady | ANO (libovolný) | Ano (Apache 2.0) |
| Cursor | 200 comp./m | Pro \$20/m, Business \$40/u | Cast. (model picker) | Ne |
| Antigravity | ANO (ToS riziko) | Zdarma (rotace Google) | Ne | Ne (MIT plugin) |
| Claude Cowork | Ne | Součást Pro \$20/m | Ne (jen Anthropic) | Ne |

Závěr: jak si vybrat nástroj

Chceš nejlepší CLI agenta, cena nerozhoduje

Claude Code — nejlepší CLAUDE.md paměťový systém, auto-memory, hooks, multi-agent orchestrace. Pro produktivní týmy: CLAUDE.md commitnutý do gitu.

Chceš zdarma bez kreditky

Gemini CLI — jediný model-lockovaný nástroj s opravdu bezplatným účtem (1000 req/den). Bonus: checkpointing (shadow git) a /rewind jsou unikátní.

Chceš BYOM a plátit jen za API

OpenCode (TUI, 75+ providerů, MIT) nebo Continue.dev (VS Code extension, open source). Antigravity plugin pro OpenCode: bezplatný Gemini + Claude, ale ToS riziko.

Chceš nejbezpečnější sandbox pro autonomní agenty

Codex CLI — platform-native Seatbelt/Linux sandbox, síť VYPNUTA výchozí. Multimodal vstup (obrázky, screenshoty, wireframy). Součást ChatGPT Plus.

Jsi neprogramátor nebo manažer

Claude Cowork — stejná síla jako Claude Code bez terminálu. Excel, PowerPoint, správa souborů, analýza dat, browser automatizace. Součást Pro plánu bez příplatku.