

# REAL-WORLD CODING WITH AI: EFFECTIVE TOOLS & WORKFLOWS



November 7, 2025

**John Riviello**

@JohnRiv

<sup>01</sup> **AI Coding Tools Landscape**

<sup>02</sup> **Useful Use Cases**

<sup>03</sup> **Prompting**

<sup>04</sup> **Spec-Driven Development (SDD)**

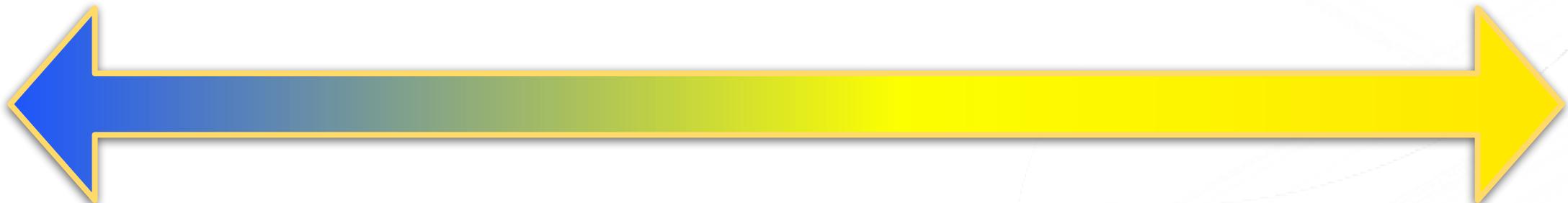
<sup>05</sup> **Model Context Protocol (MCP)**

<sup>06</sup> **Adoption & Expectations**



Hello!

# AI-ASSISTED CODING SENTIMENT SPECTRUM



**“What we are finding is that we’re 3 to 6 months from a world where AI is writing 90 percent of the code. And then in 12 months, we may be in a world where AI is writing essentially all of the code,”**

**- Dario Amodei, Anthropic CEO**

**March 2025**

**"Some people think that prediction is wrong,  
but within Anthropic and within a number of  
companies that we work with,  
that is absolutely true now."**



**"If Claude is writing 90% of the code, what that means, usually, is, you need just as many software engineers..."**

**They can focus on the 10% that's editing the code or writing the 10% that's the hardest, or supervising a group of AI models. "**



# AI-ASSISTED CODING SENTIMENT SPECTRUM



# 01 AI Coding Tools Landscape

## 02 Useful Use Cases

## 03 Prompting

## 04 Spec-Driven Development (SDD)

## 05 Model Context Protocol (MCP)

## 06 Adoption & Expectations

# AI Foundation Models

# LATEST AI FOUNDATION MODELS

## OpenAI

- GPT-5 (low reasoning)
- GPT-5 (medium reasoning)
- GPT-5 (high reasoning)
- GPT-5-Codex

## Anthropic

- Claude Sonnet 4.5
- Claude Haiku 4.5
- Claude Opus 4.1

## Google

- Gemini 2.5 Pro
- Gemini 2.5 Flash

# AI Coding Tools

# CHATBOTS THAT CAN WRITE CODE

## OpenAI

- ChatGPT

## Anthropic

- Claude

## Google

- Gemini

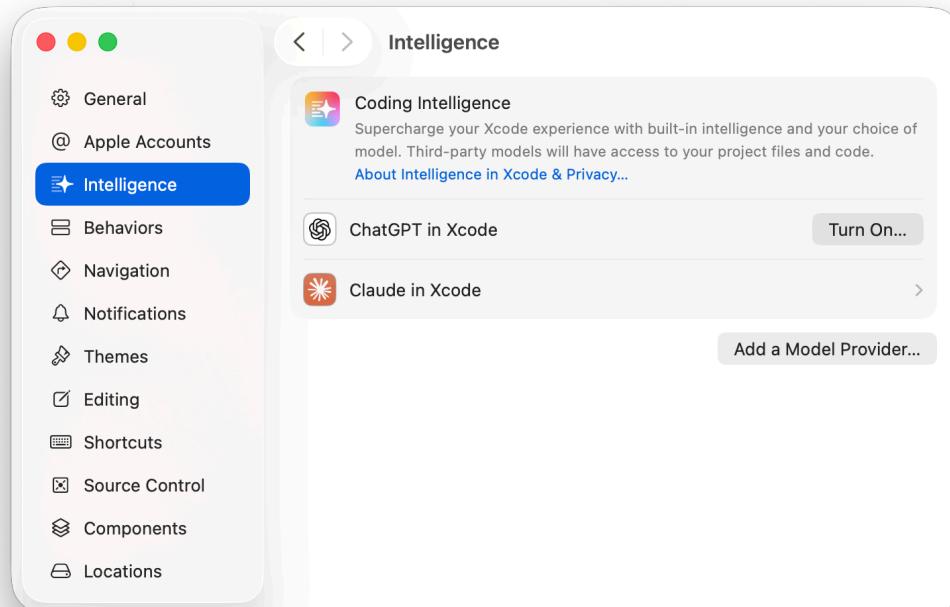
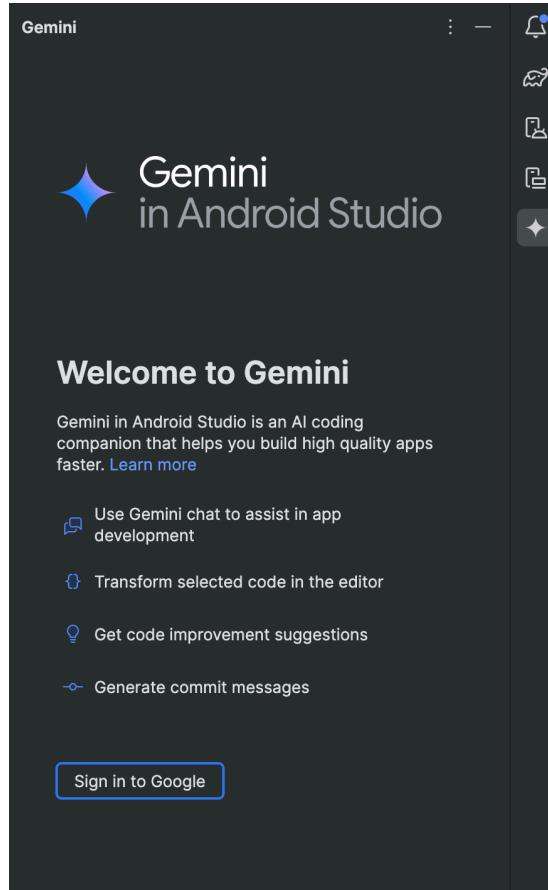
# COMMAND LINE TOOLS

- OpenAI Codex CLI
- Claude Code
- Gemini CLI
- Aider

# PLUGINS FOR IDEs

- GitHub Copilot
- RooCode
- Augment Code
- OpenAI Codex IDE Extension
- Cline
- Amp
- AI Assistant by JetBrains
- Gemini Code Assist

# IDEs THAT ADDED AI CAPABILITIES



# Xcode

# Zed

# AGENTIC IDEs

- Cursor
- Windsurf
- Kiro
- Firebase Studio

# AUTONOMOUS CODING AGENTS

- GitHub Copilot Coding Agent
- Claude Code on the web
- Jules
- OpenAI Codex Cloud
- Devin
- Junie

# AI CODE REVIEWERS

- GitHub Copilot

<https://docs.github.com/en/copilot/how-tos/use-copilot-agents/request-a-code-review/configure-automatic-review>

- CodeRabbit

<https://www.coderabbit.ai/>

- LlamaPReview

<https://jetxu-llm.github.io/LlamaPReview-site/>

# PROTOTYPE GENERATORS

- Replit
- Bolt
- v0
- GitHubSpark
- Lovable
- Firebase Studio

# AGENT CLIENT PROTOCOL (ACP)

- Standardizes communication between code editors/IDEs, and coding agents
- Began as a collaboration between Google & Zed to bring Gemini CLI into Zed
- Provides a standardized protocol for agent-editor communication, similar to how the Language Server Protocol (LSP) standardized language server integration

## RELATED LINKS

- Agent Client Protocol spec  
<https://agentclientprotocol.com/>
- Zed Blog: How the Community is Driving ACP Forward  
<https://zed.dev/blog/acp-progress-report>
- JetBrains × Zed: Open Interoperability for AI Coding Agents in Your IDE  
<https://blog.jetbrains.com/ai/2025/10/jetbrains-zed-open-interoperability-for-ai-coding-agents-in-your-ide/>

# Costs



# TOKENS



- **Context Window** is measured in tokens
- 100 tokens ≈ 60-80 text words
  - See it in action at <https://platform.openai.com/tokenizer>
  - Images, Audio & Video are calculated differently
  - Tool use (MCP) adds additional tokens
- Cost is typically more expensive for output tokens than input tokens
- Cached input tokens are less expensive
  - **Prompt caching:** the initial, identical part of a prompt is reused from a cache to reduce latency and cost for subsequent requests

# FOUNDATION MODEL COSTS PER MILLION TOKENS

Model	Input	Cached Input	Output
GPT-5	\$1.250	\$0.125	\$10.000
Claude Sonnet 4.5	≤ 200K tokens: \$3.000 >200K tokens: \$6.000	≤ 200K tokens: \$0.300 >200K tokens: \$0.600	≤ 200K tokens: \$15.000 >200K tokens: \$22.500
Claude Haiku 4.5	\$1.000	\$0.100	\$5.000
Claude Opus 4.1	\$15.000	\$1.500	\$75.000
Gemini 2.5 Pro	≤ 200K tokens: \$1.250 >200K tokens: \$2.500	≤ 200K tokens: \$0.125 >200K tokens: \$0.250	≤ 200K tokens: \$10.000 >200K tokens: \$15.000
Gemini 2.5 Flash	\$0.300	\$0.030	\$2.500

- <https://openai.com/api/pricing/>
- <https://claude.com/pricing#api>
- <https://ai.google.dev/gemini-api/docs/pricing>

- Claude & Gemini charge for cache writes as well
- Calculator & additional model pricing: <https://www.llm-prices.com/>

# EXAMPLE COSTS FOR IDES & OTHER TOOLS

Tool	Monthly Cost	Notes
GitHub Copilot	Pro: \$10	Unlimited completions; 300 premium requests
Cursor	Pro: \$20	Unlimited completions; \$20 credit for premium models
Windsurf	Pro: \$15	Unlimited completions; 500 premium requests
Gemini Code Assist	Standard: \$22.80	Unlimited completions; 1,500 model requests / day
Claude Code	Pro: \$20	During peak hours, offers at least 5x the usage per session compared to free
OpenAI Codex	Chat GPT Plus: \$20	Average users can send about 30-150 local messages or 5-40 cloud tasks every 5 hours, with a shared weekly limit.

- <https://github.com/features/copilot/plans>
- <https://cursor.com/docs/account/pricing>
- <https://windsurf.com/pricing>
- <https://codeassist.google/>
- <https://claude.com/pricing>
- <https://chatgpt.com/pricing/> & <https://help.openai.com/en/articles/11369540-using-codex-with-your-chatgpt-plan>

01 **AI Coding Tools Landscape**

02 **Useful Use Cases**

03 **Prompting**

04 **Spec-Driven Development (SDD)**

05 **Model Context Protocol (MCP)**

06 **Adoption & Expectations**

# Autocomplete

# Inline Chat

# TDD

# Refactoring

# Upgrades

# Documentation

# Commit Messages



# Conventional Commits

A specification for adding human and machine readable meaning to commit messages

Quick Summary

Full Specification

Contribute



# Code Review

# Debugging

# Understanding Latest Changes/ Entire Codebase

# Learning New Languages & Frameworks

**“Build this feature...”**

01 AI Coding Tools Landscape

02 Useful Use Cases

03 Prompting

04 Spec-Driven Development (SDD)

05 Model Context Protocol (MCP)

06 Adoption & Expectations

# How to prompt

# INLINE HELP

```
292
293 let safeName = function(s) {
  Ask or edit in context          Claude Sonnet 4 ▾ ➤ ▾
294   return s.replace(/\W/g, '_');    You, 7 years ago • it works
295 }
296
```

# INLINE HELP

292

```
293 let safeName = function(s) {
```



/explain X What does this do?

This function creates a "safe" version of a string by replacing all non-word characters with underscores.

## How it works:

- **Input:** Takes a string `s` as parameter
- **Regex pattern:** `/\W/g` matches any non-word character
  - `\W` = any character that's NOT a letter, digit, or underscore
  - `g` flag = global (replace ALL matches, not just the first)
- **Output:** Returns the string with all non-word characters replaced by

-

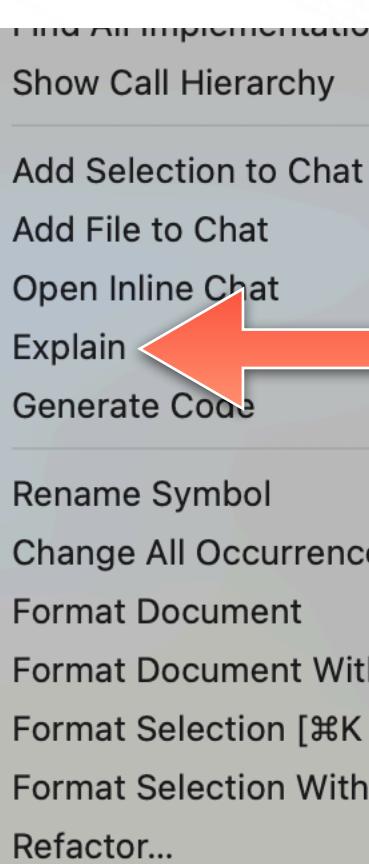
## Examples:

|Ask or edit in context

Claude Sonnet 4 ▾ ➤ ▾

# IN-FILE EXPLANATION

```
311
312  let importForm = function(event) {
313    try {
314      let data = JSON.parse(document.getElementById("datamgmt").value);
315      localStorage.clear(); // must follow spec
316      Object.keys(data).forEach(function(k) {
317        localStorage.setItem(k, data[k]);
318      });
319      window.location = window.location.pathname;
320    } catch (e) {
321      alert("JSON string invalid: "+text);
322    }
323  }
324  document.getElementById('datamgmt_import').onchange = importForm;
325
```



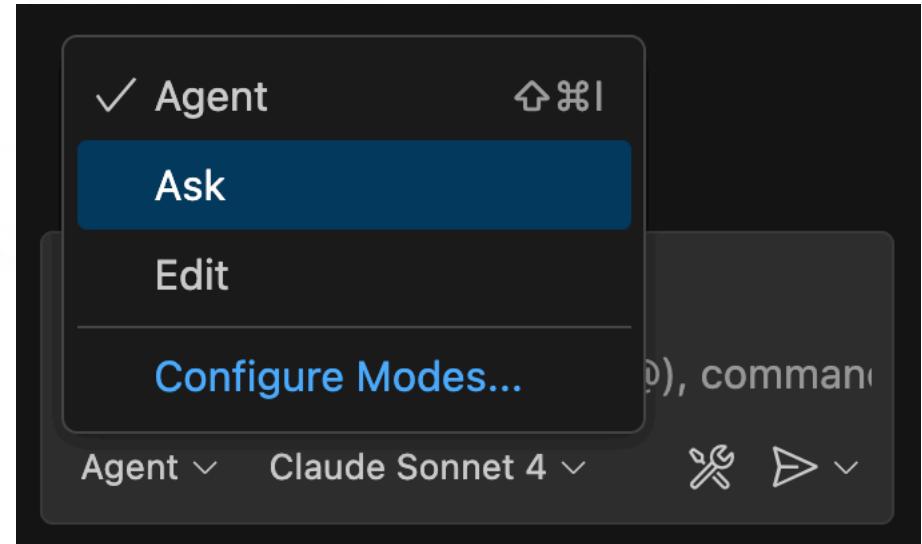
Workspace  
Used 1 reference

This code snippet performs a **data import** and **page refresh operation** that loads configuration data from a form element into the browser's local storage and then reloads the current page.

The first line retrieves data from an HTML element with the ID "datamgmt" (likely a textarea or input field), parses it from JSON format into a JavaScript object, and stores it in the `data` variable. This suggests the application has some kind of data.

# CHAT: ASK VS. EDIT VS. AGENT

- **Ask:** Quick asks
- **Edit:** Diff for you to review
- **Agent:** Let AI review your codebase, choose which files it thinks it needs to touch and/or create, and watch it do it
- Getting Started Prompts: <https://docs.github.com/en/copilot/how-tos/chat-with-copilot/get-started-with-chat>
- More Copilot-specific details: <https://copilotcheatsheet.com/>





# Context is King

# PROMPTING GUIDES

- **OpenAI**
  - <https://cdn.openai.com/API/docs/gpt-5-for-coding-cheatsheet.pdf>
  - [https://cookbook.openai.com/examples/gpt-5/gpt-5\\_prompting\\_guide](https://cookbook.openai.com/examples/gpt-5/gpt-5_prompting_guide)
- **Anthropic**
  - <https://www.anthropic.com/engineering/clause-code-best-practices>
- **Google**
  - <https://ai.google.dev/gemini-api/docs/prompting-strategies>

# WHAT WORKED TODAY, MIGHT NOT WORK TOMORROW

- “You are a Senior Software Engineer...”
- XML vs. Markdown vs. Plain Text
  - <https://maxleiter.com/blog/rewrite-your-prompts>
- Cognition had to rebuild how Devin worked when Sonnet 4.5 was released: <https://cognition.ai/blog/devin-sonnet-4-5-lessons-and-challenges>

# PLEASE



# Context is King

# GUIDING YOUR AGENTS

- **Agents.md** - <https://agents.md/>
  - README for agents: a dedicated, predictable place to provide the context and instructions to help AI coding agents work on your project.
  - Standard markdown file
  - Supported by most of the ecosystem

## HOW TO USE IT

1. Add AGENTS.md to repo root
  - Ask AI to scaffold it
2. Cover what matters
  - Project overview
  - Build and test commands
  - Code style guidelines
  - Testing instructions
  - Security considerations
3. Add extra instructions

# GUIDING YOUR AGENTS

- **How to configure Instructions / Rules**
  - **GitHub Copilot Instructions:** <https://docs.github.com/en/copilot/how-tos/configure-custom-instructions/add-repository-instructions>
  - **Cursor Rules:** <https://cursor.com/docs/context/rules>
  - **Windsurf Rules:** <https://docs.windsurf.com/windsurf/cascade/memories#rules>
- If your tool doesn't support this, or you want more control, you can always create them as markdown files & include them as context: <https://medium.com/ngconf/universal-knowledge-base-for-ai-2da5748f396c>



# Context is King

# GUIDING YOUR AGENTS

- **Resources for Pre-Built Instructions / Rules**
  - GitHub Copilot:
    - <https://github.com/github/awesome-copilot>
  - Cursor:
    - <https://github.com/PatrickJS/awesome-cursorrules>
    - <https://cursor.directory/>
    - Generator: <https://cursor.directory/generate>
  - Windsurf:
    - <https://windsurf.com/editor/directory>

# GUIDING YOUR AGENTS

- **Reusable Workflows**
  - GitHub Copilot:
    - <https://code.visualstudio.com/docs/copilot/customization/prompt-files>
  - Cursor:
    - <https://cursor.com/docs/agent/chat/commands>
  - Windsurf:
    - <https://docs.windsurf.com/windsurf/cascade/workflows>

Life is sharing



01 **AI Coding Tools Landscape**

02 **Useful Use Cases**

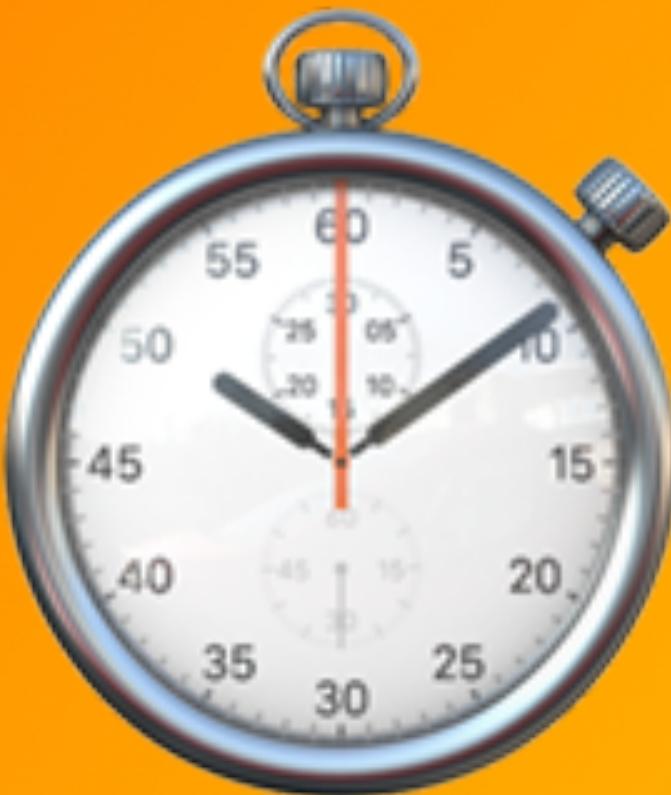
03 **Prompting**

04 **Spec-Driven Development (SDD)**

05 **Model Context Protocol (MCP)**

06 **Adoption & Expectations**

**“Build this feature...”**



# GENERAL SDD FLOW

- Build a spec
- Build an architecture plan
- Break it down into user stories/tasks
- Execute on them
- Rinse & repeat

# OPENSPEC

<https://openspec.dev/>

## Workflow:

- Draft the Proposal
- Verify & Review
- Refine the Specs
- Implement the Change
- Archive the Completed Change

```
vim2 -zsh
> openspec view
OpenSpec Dashboard

Summary:
  ● Specifications: 10 specs, 64 requirements
  ● Active Changes: 3 in progress
  ● Completed Changes: 4
  ● Task Progress: 30/41 (73% complete)

Active Changes
  ● make-validation-scope-aware [██████████] 0%
  ● remove-diff-command [██████████] 90%
  ● improve-deterministic-tests [██████████] 92%

Completed Changes
  ✓ add-slash-command-support
  ✓ sort-active-changes-by-progress
  ✓ update-agent-file-name
  ✓ update-agent-instructions

Specifications
  cli-archive          10 requirements
  cli-validate         10 requirements
  cli-list             9 requirements
  cli-view             7 requirements
  cli-init             7 requirements
  cli-update           5 requirements
  cli-change           5 requirements
  cli-spec              4 requirements
  cli-show              4 requirements
  cli-show              3 requirements
```

# MULTI-AGENT SWARMS: TOOLS

- **Claude Squad**

- <https://smtg-ai.github.io/clause-squad/>
- Open source, works with Claude Code, Codex, Gemini, and local agents

- **Conductor**

- <https://conductor.build/>
- Only Mac & Claude Code, waitlist for Windows

- **Magnet**

- <https://www.magnet.run/>
- Only Mac & Claude Code

# MULTI-AGENT SWARMS: HOWTO

## Links for when you're ready

- Launch Multiple Cursor Composer AI Agents to Work in Parallel  
<https://egghead.io/launch-multiple-cursor-composer-ai-agents-to-work-in-parallel~y1q56>
- I Managed a Swarm of 20 AI Agents for a Week and Built a Product. Here Are the 8 Rules I Learned.  
<https://zachwills.net/i-managed-a-swarm-of-20-ai-agents-for-a-week-here-are-the-8-rules-i-learned/>
- Beads: lightweight memory system for coding agents - <https://github.com/steveyegge/beads>
  - Also check out Steve's articles about Beads at <https://steve-yegge.medium.com/>
  - Conductors to Orchestrators: The Future of Agentic Coding  
<https://addyo.substack.com/p/conductors-to-orchestrators-the-future>

*Remember you still need to review all that code!*

01 **AI Coding Tools Landscape**

02 **Useful Use Cases**

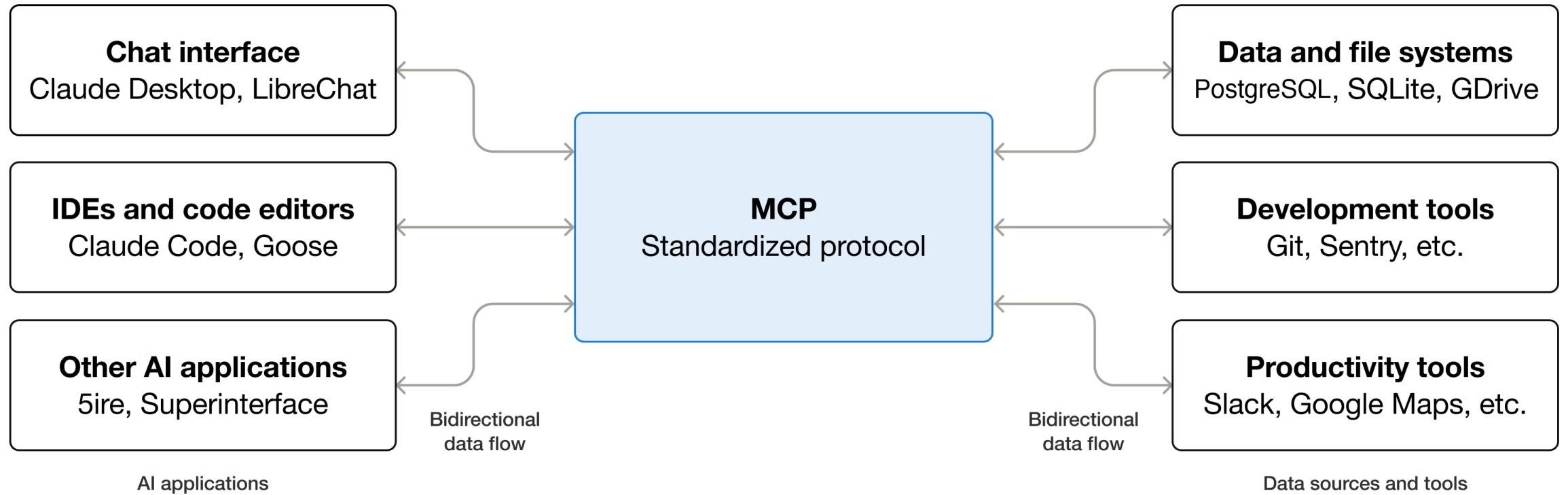
03 **Prompting**

04 **Spec-Driven Development (SDD)**

05 **Model Context Protocol (MCP)**

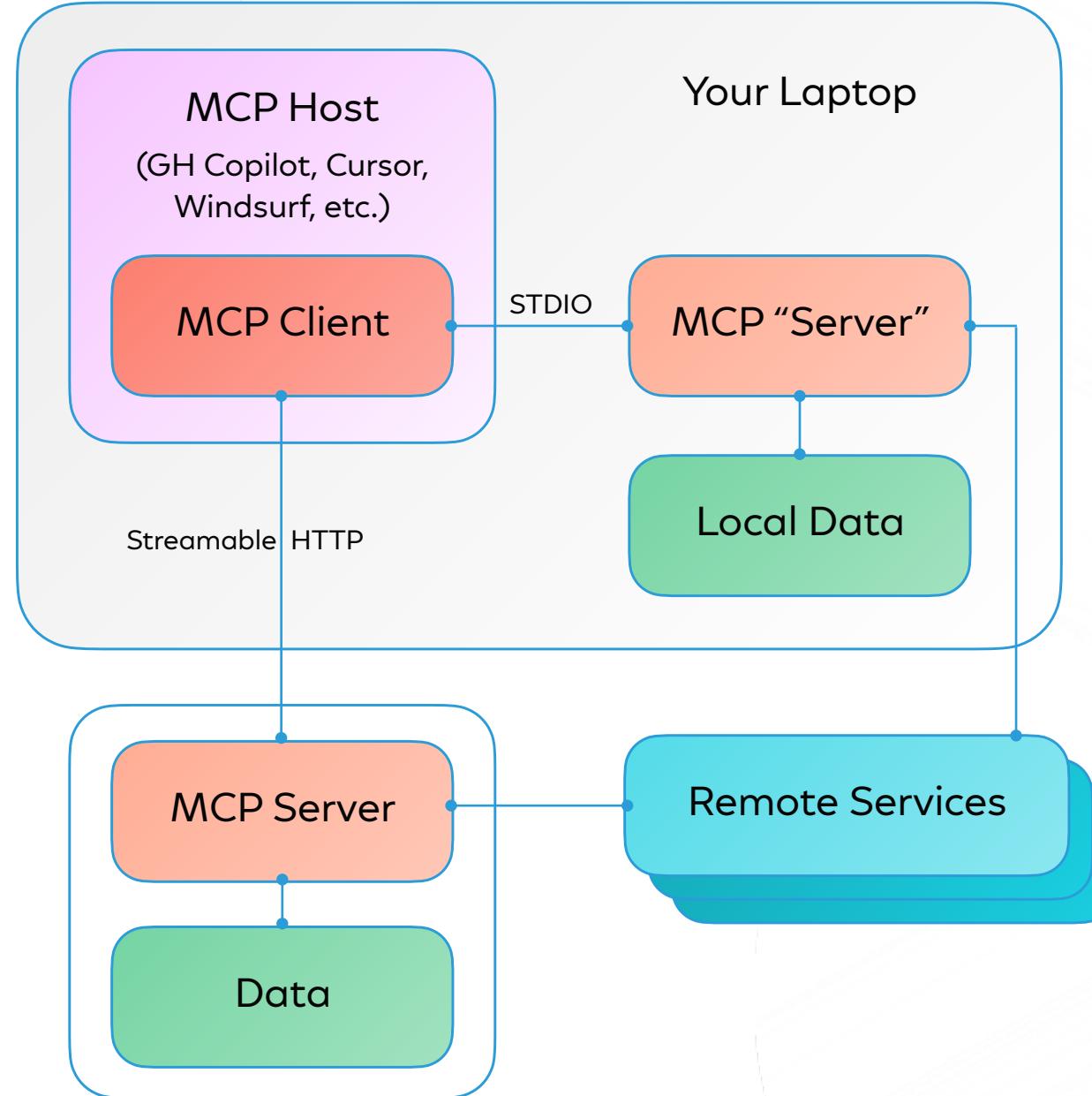
06 **Adoption & Expectations**

# MCP: WHAT & WHY



<https://modelcontextprotocol.io/>

# MCP: HOW



# USEFUL MCPS

**Markitdown**  
[Install ▾](#)  
Convert various file formats (PDF, Word, Excel, images, audio) to Markdown.  
By microsoft ⭐ 82,151

**Context7**  
[Install ▾](#)  
Get up-to-date, version-specific documentation and code examples from any library or framework.  
By upstash ⭐ 35,139

**GitHub**  
[Install ▾](#)  
Official GitHub MCP Server that connects AI tools directly to GitHub's platform. Enables AI agents to read...  
By github ⭐ 23,949

**Playwright**  
[Install ▾](#)  
Automate web browsers using accessibility trees for testing and data extraction.  
By microsoft ⭐ 22,427

**Serena**  
[Install ▾](#)  
Semantic code retrieval & editing tools for coding agents.  
By oraios ⭐ 14,878

**Chrome DevTools**  
[Install ▾](#)  
Lets your coding agent (such as Gemini, Claude, Cursor or Copilot) control and inspect a live Chrome browser.  
By ChromeDevTools ⭐ 12,852

- Most-starred, official MCP servers: <https://github.com/mcp>
- List of MCP Lists: <https://github.com/bh-rat/awesome-mcp-enterprise#mcp-directories--marketplaces>

# AUTHORIZATION

- Docs: <https://modelcontextprotocol.io/docs/tutorials/security/authorization>
- Specification: <https://modelcontextprotocol.io/specification/2025-06-18/basic/authorization>
- Authorization for remote servers: follows OAuth 2.1 conventions
- Authorization for local servers: typically a key in local configuration
- Important Considerations: [https://modelcontextprotocol.io/specification/2025-06-18/basic/security\\_best\\_practices](https://modelcontextprotocol.io/specification/2025-06-18/basic/security_best_practices)

# The Lethal Trifecta



**What to do if you are  
blocked from using MCP**

## Up-to-date Docs for LLMs and AI code editors

Copy latest docs & code — paste into Cursor, Claude, or other LLMs

Search a library (e.g. Next, React)

Popular

Trending

Recent

	SOURCE	TOKENS	SNIPPETS	UPDATE	
Next.js	🔗 /vercel/next.js	630K	3.3K	2 hours	✓
MongoDB	🔗 /mongodb/docs	15.8M	131K	1 week	✓
Shadcn UI	🔗 /shadcn-ui/ui	215K	1.3K	1 day	✓
Vercel AI SDK	🔗 /vercel/ai	559K	2.4K	2 hours	✓
Better Auth	🔗 /better-auth/better-auth	249K	1.4K	6 days	✓
Supabase	🔗 /supabase/supabase	1.2M	4.7K	4 days	✓
React	🔗 /reactjs/react.dev	637K	2.4K	1 week	✓
LangGraph	🔗 /langchain-ai/langgraph	635K	2K	1 day	✓
Tailwind CSS	🌐 tailwindcss.com/docs	240K	1.6K	4 days	✓
Tailwind CSS	🔗 /tailwindlabs/tailwindcss.com	565K	1.8K	2 weeks	✓

<https://context7.com/>

[The /llms.txt file](#)[Code](#)[Python module & CLI](#)[Python source](#)[JavaScript](#)[Editors and IDEs](#)[ed, the standard text editor](#)[Tutorials](#)[llms.txt in Different Domains](#)[How to help LLMs understand your nbdev project](#)

# The /llms.txt file

A proposal to standardise on using an [/llms.txt](#) file to provide information to help LLMs use a website at inference time.

**AUTHOR**

Jeremy Howard

**PUBLISHED**

September 3, 2024

On this page

[Background](#)[Proposal](#)[Format](#)[Edit](#)<https://llmstxt.org/>

## Background

Large language models increasingly rely on website information, but face a critical limitation: context windows are too small to handle most websites in their entirety. Converting complex HTML pages with navigation, ads, and JavaScript into LLM-friendly plain text is both difficult and imprecise.

While websites serve both human readers and LLMs, the latter benefit from more concise, expert-level information gathered in a single, accessible location. This is particularly important for use cases like development environments, where LLMs need quick access to programming documentation and APIs.

## Proposal

We propose adding a [/llms.txt](#) markdown file to websites to provide LLM-friendly content. This file offers brief background information, guidance, and links to detailed markdown files.

llms.txt markdown is human and LLM readable, but is also in a precise format allowing fixed processing methods (i.e. classical programming techniques such as parsers and regex).

We furthermore propose that pages on websites that have information that might be useful for LLMs to read provide a clean markdown version of those pages at the same URL as the original page, but with `.md` appended. (URLs without file names should append



llms.txt logo

# MCP REGISTRY



A screenshot of a GitHub repository page for 'MCP Registry'. The top navigation bar includes links for 'README' (highlighted with a red underline), 'Code of conduct', 'Contributing', 'MIT license', 'Security', and edit/pencil icons.

## MCP Registry

The MCP registry provides MCP clients with a list of MCP servers, like an app store for MCP servers.

[!\[\]\(41afd5b1f1c6ce5ca3de56a2811164e8\_img.jpg\) Publish my MCP server](#) | [!\[\]\(88f68fd6a207f986b29d243c02aeb11a\_img.jpg\) Live API docs](#) | [!\[\]\(8afe7c5303eddb723f7434703180fbcb\_img.jpg\) Ecosystem vision](#) | [!\[\]\(3317c0d1d01cac35dd613b456fb3fea3\_img.jpg\) Full documentation](#)

## Development Status

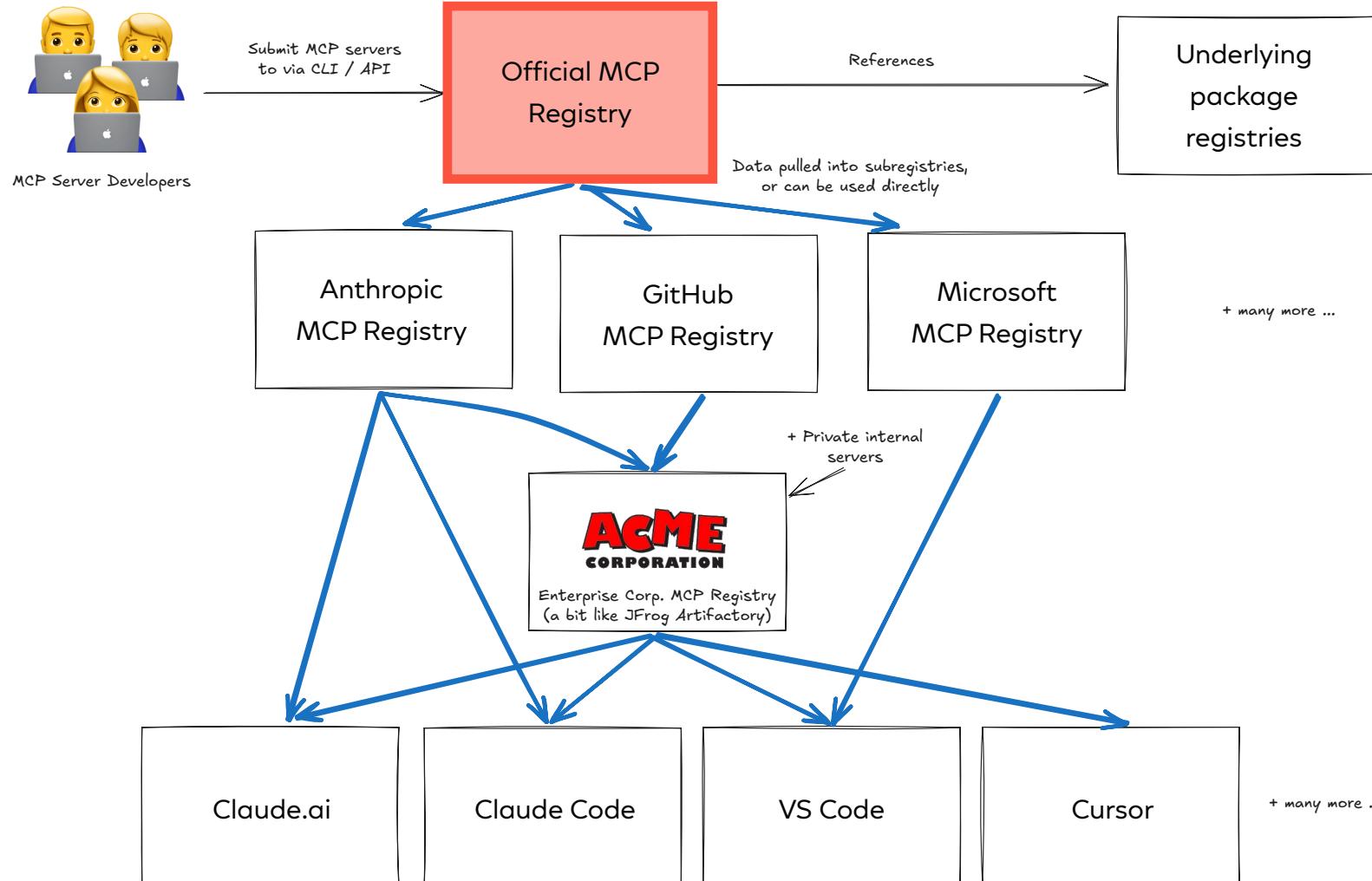
**2025-10-24 update:** The Registry API has entered an [API freeze \(v0.1\)](#) 🎉. For the next month or more, the API will remain stable with no breaking changes, allowing integrators to confidently implement support. This freeze applies to v0.1 while development continues on v0. We'll use this period to validate the API in real-world integrations and gather feedback to shape v1 for general availability. Thank you to everyone for your contributions and patience—your involvement has been key to getting us here!

**2025-09-08 update:** The registry has launched in preview 🎉 ([announcement blog post](#)). While the system is now more stable, this is still a preview release and breaking changes or data resets may occur. A general availability (GA) release will follow later. We'd love your feedback in [GitHub discussions](#) or in the [#registry-dev Discord \(joining details here\)](#).

# MCP REGISTRY ECOSYSTEM VISION

- **2 parts to the registry project:**
  - **The MCP registry spec:** An API specification that allows anyone to implement a registry.
  - **The Official MCP registry:** A hosted registry following the MCP registry spec at [registry.modelcontextprotocol.io](https://registry.modelcontextprotocol.io)
    - Serves as the authoritative repository for publicly-available MCP servers.
    - Server creators publish once, and all consumers (MCP clients, aggregators, marketplaces) reference the same canonical data.
    - Owned by the MCP open-source community and backed by major trusted contributors to the MCP ecosystem such as Anthropic, GitHub, PulseMCP & Microsoft.

# MCP REGISTRY ECOSYSTEM VISION





# Context is King

**3-4 MCP Tools**

**\*AT MOST\***

01 **AI Coding Tools Landscape**

02 **Useful Use Cases**

03 **Prompting**

04 **Spec-Driven Development (SDD)**

05 **Model Context Protocol (MCP)**

06 **Adoption & Expectations**

# AI-ASSISTED CODING SENTIMENT SPECTRUM

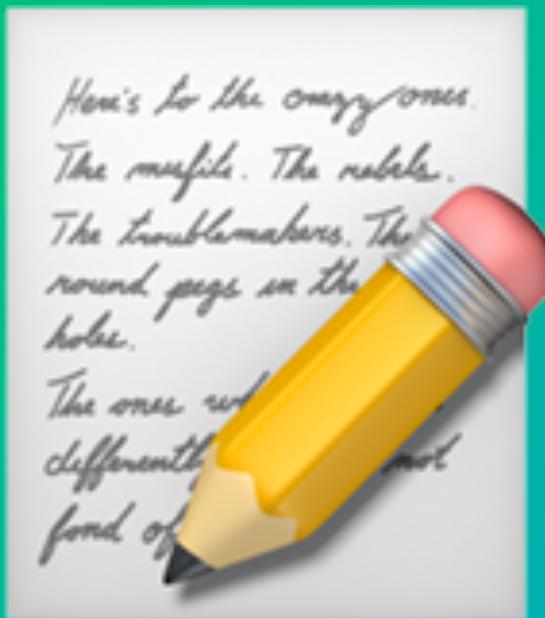


# Top-down Support

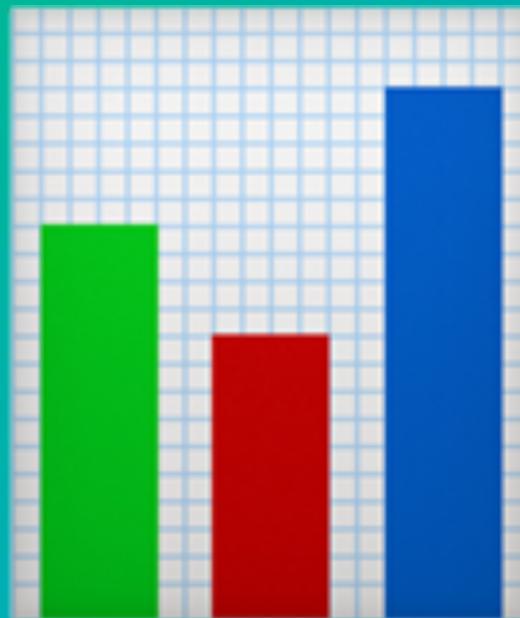
# Bottom-up Champions







Here's to the crazy ones.  
The misfits. The rebels.  
The troublemakers. The  
round pegs in the  
square hole.  
The ones who  
see things  
differently.  
They're not  
fond of  
rules.









# THE 70% PROBLEM