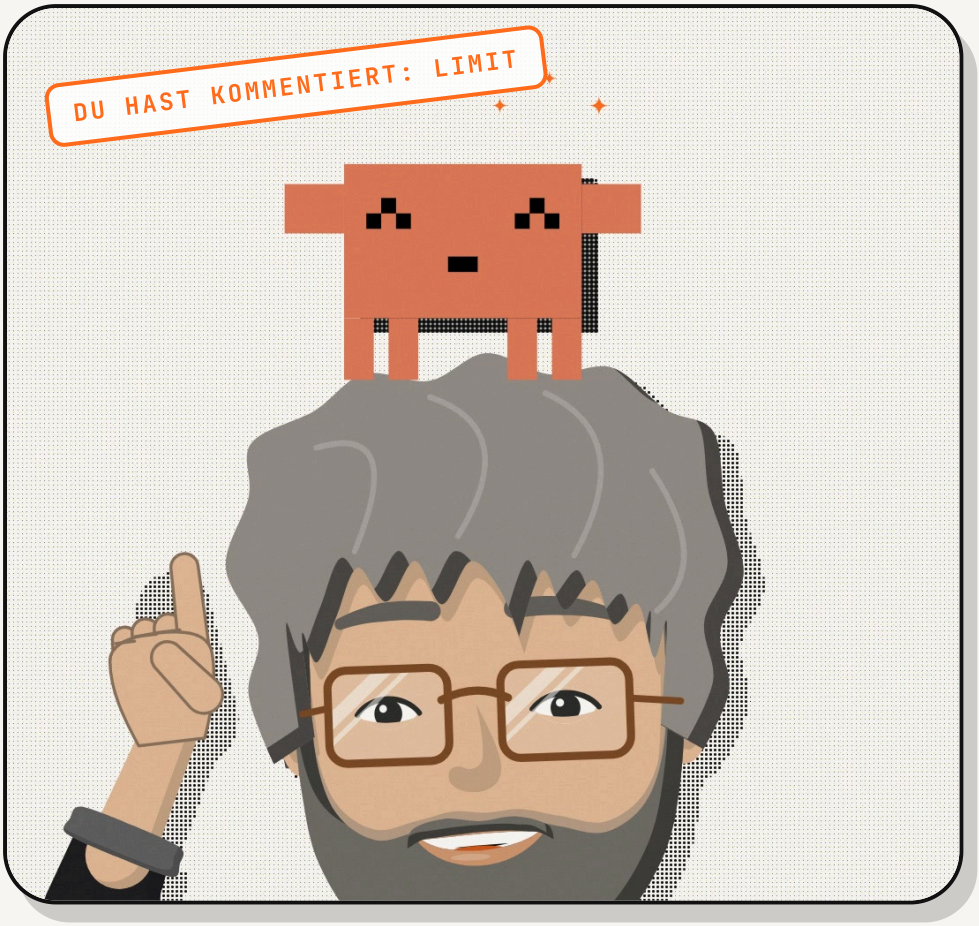


Was die Arena dein *Limit kostet*

Die ganze Rechnung hinter dem Video: was der Skill wirklich macht, was ein Tester gemessen hat, wie die 39 Millionen Tokens zustande kommen, warum dein Pro-Plan das nicht trägt und wie du jeden Hype-Skill prüfst, bevor er dein Limit frisst. Mit allen Quellen.

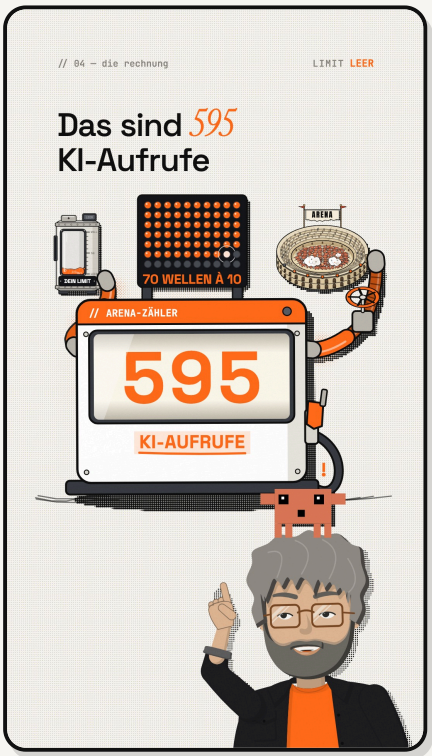
// 01

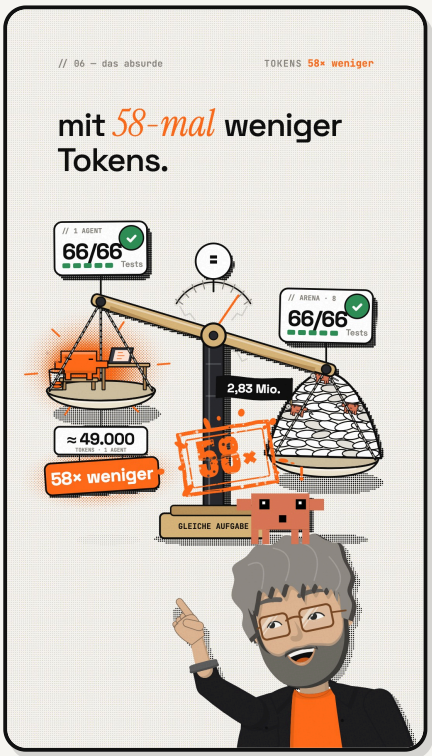
Auf einen Blick

FRAGE	KURZ GESAGT
Was ist Arena?	Ein Community-Skill auf GitHub (Jakeschincariol/arena-skill), kein Feature von Anthropic
Was macht er?	Startet standardmäßig 100 Sub-Agenten, die sich gegenseitig angreifen, bis eine Antwort übrig ist
Wie viele Aufrufe?	595 KI-Aufrufe in 70 Wellen zu je 10, für eine einzige Aufgabe (laut README)
Gemessen	8 Agenten = 43 Aufrufe ≈ 2,83 Mio. Tokens (ein Tester, ein Test)
Hochgerechnet	100 Agenten ≈ 39 Mio. Tokens (geschätzt, linear hochgerechnet)
Das Ergebnis	Ein einzelner Agent bestand dieselben 66 Tests, mit etwa 58-mal weniger Tokens
Dein Plan	Pro und Max haben ein Fünf-Stunden- und ein Wochenlimit, geteilt mit dem Claude-Chat

// AUS DEM VIDEO







// 02

Was dir die Hype-Posts nicht sagen

Arena ist echt, und die Idee klingt stark: 100 Versionen von Claude bekommen dieselbe Aufgabe, jede mit einer anderen Strategie-Karte, sie greifen sich paarweise an, ein Richter wertet, der Verlierer fliegt raus. 100 werden 50, dann 25, 13, 7, 4, 2, 1. Was in den Posts fehlt:

- **Es ist kein Anthropic-Feature.** Der Skill stammt von einem einzelnen Entwickler, steht unter MIT-Lizenz auf GitHub und ist nicht Teil der offiziellen Skills von Anthropic. Niemand bei Anthropic hat ihn geprüft.
- **Der Standard ist das Maximum.** Ohne Zusatz startet `/arena` mit 100 Agenten. Laut der Anleitung im Skill nennt Claude kurz die Größe („100 agents, 7 rounds, 595 sub-agent calls“) und legt los, ohne zu fragen.
- **Er kann sich selbst melden.** Der Skill ist so beschrieben, dass Claude ihn auch ohne Befehl vorschlagen kann, etwa wenn du „versuch es nochmal“ oder „schlechte Antwort“ schreibst. In dem Fall soll er vorher fragen. Lies die Frage dann genau.
- **Hunderte Freigaben.** Jeder Sub-Agent schreibt eine Datei in den Ordner `.arena/`. Im normalen Modus heißt das: eine Freigabe pro Datei, bei einem großen Lauf Hunderte.
- **Es dauert.** Claude Code führt standardmäßig höchstens 10 Werkzeug-Aufrufe gleichzeitig aus. Darum läuft die Arena in Wellen, 70 Stück beim vollen Lauf.



DER MENTOR SAGT

Ein Skill mit 1.000 Sternen ist beliebt. Das sagt nichts darüber, was er dich pro Prompt kostet.

// 03

Die ganze Rechnung

Der Entwickler veröffentlicht die Zahl der Aufrufe selbst, im README unter „What it costs“. Die Tokens pro Aufruf stehen dort nicht, denn sie hängen von deiner Aufgabe ab: „Every call reads the task and one or two solutions, so the bill grows with the size of the task.“

AGENTEN	RUNDEN	AUFRUFE	WELLEN À 10
100 (Standard)	7	595	70
64	6	379	49
32	5	187	28
16 (`--quick`)	4	91	16
8	3	43	10
4	–	19	–

Quelle: README des Repos, die Zeile mit 4 Agenten aus der Messung in Issue #1.

Was ein Tester gemessen hat

In Issue #1 hat ein Nutzer (mersad31) die Kosten gemessen, bevor er den Skill für echte Arbeit genutzt hat. Seine Aufgabe: ein Rechenprogramm in Python mit klaren Sicherheitsregeln, dazu 66 versteckte Tests und ein blinder Richter.

LAUF	AUFRUFE	TOKENS
Ein einzelner Agent	1	≈ 49.000
Arena mit 8 Agenten	43	≈ 2,83 Mio.

Pro Aufruf sind das rund 66.000 Tokens. Die teuersten Phasen waren Angriff und Verteidigung, zusammen etwa 65 % der Rechnung.

So kommen die 39 Millionen zustande

Die Rechnung ist einfach und ausdrücklich eine Schätzung: 2,83 Mio. Tokens für 43 Aufrufe, hochgerechnet auf 595 Aufrufe, ergibt rund 39 Mio. Tokens. Für `--quick` (91 Aufrufe) kommt der Tester auf rund 6 Mio. Das gilt für genau diese Aufgabe. Eine längere Aufgabe kostet mehr, eine sehr kurze weniger.

Zum Vergleich: Der KI-Rat aus meinem nächsten Video (roast-room) nennt in seiner eigenen Anleitung für den vollen Rat 21 Sub-Agent-Läufe und „about 1M tokens“.

Was dabei herauskam

Der einzelne Agent und der Arena-Sieger bestanden beide alle 66 von 66 Tests. Für Aufgaben, die ein einzelner Agent schon gut löst, kostete die Arena laut Tester etwa das 58-Fache für dasselbe funktionale Ergebnis. Einen Vorteil fand er bei einem bösartigen Sonderfall, den der Arena-Sieger sauber abfing. Dazu kommt: 6 von 8 Lösungen landeten beim selben Ansatz, und du bekommst am Ende nur eine Antwort zurück.

// 04

Warum dein Pro-Plan das nicht trägt

Pro und Max haben laut Anthropic zwei Grenzen: ein Limit pro fünf Stunden und ein Wochenlimit. Beide gelten gemeinsam für Claude im Browser, in der App und in Claude Code. Alles, was du dort tust, zieht vom selben Topf ab, auch jeder Sub-Agent: „The subagent's own requests still draw on your usage“, schreibt die Claude-Code-Doku.

Wie viele Tokens in einen Pro-Plan passen, veröffentlicht Anthropic nicht. Eine exakte Rechnung „Pro = X Tokens“ gibt es deshalb nicht, und wer dir eine zeigt, schätzt. Was es gibt, sind zwei klare Hinweise:

1. **Der Nutzerbericht in Issue #1:** Ein Prompt, 16 Minuten Laufzeit, danach waren alle Tokens weg, und eine Antwort kam nie an.
2. **Die Größenordnung:** Ein einzelner Agent brauchte im Test rund 49.000 Tokens. Die volle Arena liegt geschätzt beim 800-Fachen davon.

Wenn das Limit mitten im Lauf greift, steht das Turnier still. Die Wellen davor hast du trotzdem bezahlt. Darum mein Satz im Video: Mit einem Pro-Plan ist dein Limit leer, bevor die Arena fertig ist.



DER MENTOR SAGT

Das Gefährliche an diesem Skill ist nicht die Idee. Es ist der Standardwert 100 und dass er ohne Rückfrage startet.

// 05

Dein Limit im Blick

WO	WAS DU SIEHST
`/usage` in Claude Code	Balken für dein Plan-Limit, dazu eine Aufschlüsselung, welcher Anteil auf Skills, Sub-Agenten, Plugins und MCP-Server ging (Taste `d` = 24 Stunden, `w` = 7 Tage)
`/status` in Claude Code	dein verbleibendes Kontingent, laut Hilfe-Center von Anthropic
claude.ai → Einstellungen → Nutzung	dein Limit im Browser, dazu Nutzungs-Guthaben und ein monatliches Ausgabelimit
Warnungen	Claude meldet sich, wenn du dich einem Limit näherst

Claude Code · Kosten verwalten

die offizielle Doku zu `/usage`, Sub-Agenten und Token-Verbrauch

code.claude.com ↗

Claude Code mit Pro oder Max

Fünf-Stunden- und Wochenlimit, was am Limit passiert

support.claude.com ↗

Mein Rat: Wenn du Nutzungs-Guthaben (Usage Credits) aktiviert hast, mit denen du über dein Plan-Limit hinaus weiterarbeiten kannst, setz in claude.ai unter Einstellungen → Nutzung ein monatliches Ausgabelimit, bevor du irgendeinen Multi-Agenten-Skill ausprobierst. Ohne gesetztes Limit zeigt `/usage` dort „Unlimited“.

// 06

So prüfst du jeden Skill, bevor du ihn installierst

Dafür brauchst du keine Programmierkenntnisse. Fünf Minuten reichen:

1. **Wer steht dahinter?** Offizieller Anbieter oder eine Einzelperson? Steht der Skill bei Anthropic oder nur in einem privaten Repo?
2. **Gibt es einen Kosten-Abschnitt?** Gute Skills sagen selbst, wie viele Aufrufe sie machen. Arena tut das sogar. Lies ihn.
3. **Wie viele Sub-Agenten startet er?** Such in der `SKILL.md` nach „Agent“, „sub-agent“, „parallel“ oder „wave“. Jeder Sub-Agent zieht von deinem Limit ab.
4. **Was ist der Standardwert?** Startet er mit dem Maximum oder mit einer kleinen Größe? Fragt er vor dem Start?
5. **Wann startet er sich selbst?** Die Beschreibung im Kopf der `SKILL.md` entscheidet, bei welchen Sätzen Claude ihn von allein vorschlägt.
6. **Was sagen die Issues?** Klick auf „Issues“ im Repo. Beschwerden über Kosten, Endlosschleifen oder leere Limits stehen dort zuerst.
7. **Klein testen, dann messen.** Erst eine Mini-Aufgabe, dann `/usage` öffnen und nachsehen, wie viel Prozent sie gekostet hat.

Lass Claude den Skill prüfen, ohne ihn zu installieren

```
Installiere nichts. Lies dieses GitHub-Repo: [Repo-Link]. Öffne README und SKILL.md und sag mir: 1. Wie viele Sub-Agenten oder Modell-Aufrufe startet der Skill pro Lauf im Standard? 2. Welche Optionen machen ihn kleiner? 3. Fragt er vor dem Start, oder legt er sofort los? 4. Bei welchen Sätzen startet er sich selbst? 5. Was steht in den offenen Issues zu Kosten oder Limits? Antworte mit einer kurzen Tabelle und einer klaren Empfehlung für einen Pro-Plan.
```

```
Ich habe einen Pro-Plan. Bevor ich den Skill [Name] benutze: Schätze grob, wie viel meines Fünf-Stunden-Limits ein Lauf mit den Standardwerten kosten könnte, und sag mir ehrlich, wenn du es nicht wissen kannst. Schlag mir dann die kleinste Einstellung vor, mit der ich ihn testen kann.
```

// 07

Wenn du Arena trotzdem testen willst

Dann nie mit dem Standard. Lass dir vorher die Größe zeigen, das schreibt nichts und kostet nichts:

```
python3 skills/arena/bracket.py plan --agents 4
```

Der Befehl zeigt Runden, Aufrufe und Wellen für die gewählte Zahl. Mit `/arena --agents 4` sind es laut Messung 19 Aufrufe, mit `/arena --quick` (16 Agenten) 91. Halte die Aufgabe kurz und eindeutig, denn jeder Aufruf liest sie neu. Danach `/usage` öffnen und nachsehen, was der Lauf gekostet hat.

Jakeschincariol/arena-skill

README mit Kosten-Tabelle und Anleitung

github.com ↗

Issue #1 · Messung und Nutzerbericht

2,83 Mio. Tokens, 66/66 Tests, Hochrechnung

github.com ↗

Pull Request #2

Vorschlag: Token-Schätzung zeigen und vor großen Läufen fragen (am 27.10.2026 noch nicht übernommen)

github.com ↗

// 08

Checkliste vor dem nächsten Hype-Skill

- [ ] Ist der Skill offiziell oder von einer Einzelperson?
- [ ] Kosten-Abschnitt im README gelesen
- [ ] Zahl der Sub-Agenten im Standard gefunden
- [ ] Geprüft, ob er vor dem Start fragt
- [ ] Offene Issues zu Kosten und Limits gelesen
- [ ] Ausgabelimit für Nutzungs-Guthaben gesetzt
- [ ] Erst eine Mini-Aufgabe, dann `/usage` geprüft
- [ ] Sterne und Views nicht mit Qualität verwechselt

// 09

Mein Tipp

Mehr Agenten heißt nicht mehr Wahrheit. Es heißt vor allem mehr Verbrauch. Bevor du einen Skill installierst, der „100 Claudes“ verspricht, stell eine Frage: Würde ein einzelner, gut gebriefter Claude das auch schaffen? Meistens ja. Und wenn du mehrere Sichtweisen willst, gibt es Wege, die mit einem Bruchteil der Aufrufe auskommen. Einen davon zeige ich dir im nächsten Video: einen KI-Rat mit rund 20 Aufrufen statt 595.

Alle Zahlen, Links und Befehle am 27.10.2026 in den Originalquellen geprüft: im README und in der SKILL.md des Repos, in Issue #1 und Pull Request #2, in der Claude-Code-Doku und im Hilfe-Center von Anthropic. Geschätzte Werte sind als geschätzt gekennzeichnet.



DER MENTOR SAGT

Folge mir, dann erreichen dich solche Warnungen, bevor der nächste Hype dein Limit frisst.

— @reposprites

Du willst auch so animierte Videos machen? Dann besuch meine Webseite und hol dir die Reel-Engine.

reposprites.com ↗