

Code with Claude טוקיו 2026:

הרצאת הפתיחה

שמואל לזקס, מנכ"ל Claude Code ו-Mythos 5, Claude Managed Agents ו-Fable 5: שלוש שכבות שמכוננות לאותו סיפור, לסגור את הפער בין מה שהמודל יכול לעשות לבין מה שהוא באמת עושה

12 · Claude · Kat Wu, Angela Jiang, Diane Penn, Katelyn Lesse ביוני 2026

משך: כ-42 דקות · YouTube

סיכום אוטומטי — AI Briefing (aibriefing.dev)

על המסמך הזה

זו הרצאת הפתיחה של Code with Claude בטוקיו, הפעם הראשונה שהכנס מתקיים ביפן. עולים בה ארבעה דוברים מ-Anthropic. Katelyn Lesse-ראש ה-engineering של Claude Platform, פותחת ומציגה את התמונה הכללית. Diane Penn, ראש ניהול המוצר במחקר, מציגה את המודלים החדשים. Angela Jiang, ראש המוצר של הפלטפורמה, ו-Katlyn Wu מציגות יחד את Claude Managed Agents. ראש המוצר של Claude Code, סוגרת עם החידושים ב-Claude Code. הרקע הוא השקה שקרתה, לדברי הדוברים, כמה שעות לפני ההרצאה: הדור החמישי של המודלים, Claude Fable 5 ו-Claude Mythos 5.

הטענה המרכזית: יכולות המודלים עולות באקספוננט, אבל רוב העסקים עדיין מתקדמים באופן לינארי, ולכן הפער בין מה ש-AI יכול לעשות לבין מה שהוא עושה בפועל הולך וגדל. Anthropic מציגה את עצמה כחברת פלטפורמה. לפי ההרצאה, הדרך לסגור את הפער היא שלוש שכבות: מודל חזק יותר, תשתית agents מנוהלת (harness, context, infrastructure) וכלי פיתוח שמאפשרים למפתח להפעיל צי של agents במקביל.

איך לקרוא את המסמך

- **1. תקציר מנהלים:** הטענות המרכזיות, הראיות ומה שלא נאמר.
- **2. פלטפורמה וחזון:** פער האקספוננט, נתוני השימוש ולקוחות לדוגמה.
- **3. המודלים:** Fable 5, Mythos 5, מנגנון ה-safeguards והמלצות לבנייה לקראת המודל הבא.
- **4. Claude Managed Agents:** harness, context, infrastructure, הדמו של Shinkiro Racing והפיצ'רים החדשים.
- **5. Claude Code:** ממשקים חדשים, Claude Security, routines ו-dynamic workflows, כולל דמו לוקליזציה.
- **6-7. מה עושים עכשיו ומילון מונחים:** צעדים מעשיים, שאלות פתוחות וטבלת מונחים.

הערות מקור המסמך הופק אוטומטית מהתמלול ומהשקפים של ההקלטה. שמות ומונחים עלולים להיות משובשים בתמלול האוטומטי (ASR). למשל, "Claude" מופיע לא פעם כ-"Cloud" או "Quad", ושמות חלק מהדוברים לא אומתו. הציטוטים מועתקים מילה במילה ונבדקו מול חותמת הזמן שלהם. לא נוסף מידע ממקורות חיצוניים. שמות המודלים והמספרים מובאים כפי שנאמרו על הבמה.

1. תקציר מנהלים

ההרצאה היא בעיקר אירוע השקה: מודל חדש שזמין לכולם, פיצ'רים חדשים ב-Managed Agents ומשטחי עבודה חדשים ב-Claude Code. מתחת לשיווק יש קו מחשבה עקבי: המודלים כבר מסוגלים לעבוד באופן אוטונומי לאורך זמן, ועכשיו צריך לבנות את התשתית ואת הארגון שיתנו להם לעשות את זה.

- **Fable 5 הוא המודל הכי חזק ש-Antropic הפכה לזמין לכולם:** לפי הדוברים הוא מוביל ב-SWE-Bench Pro, והפער מול מודלים אחרים גדל ככל שהמשימה ארוכה ומורכבת יותר. הם מייחסים את זה לנכונות כבר בניסיון הראשון ולאוטונומיה לאורך זמן [10:00–11:07].

- **Mythos 5 הוא אותו מודל בלי ה-safeguards:** "the same underlying model as Fable 5, but with the cyber and bio safeguards lifted". הוא זמין רק לשותפי Project Glasswing, ובהמשך החודש תיפתח הרשמה לחוקרים ממדעי החיים [15:47].

- **מנגנון ה-safeguards מנתב ל-Opus 4.8 ומחייב לפי תעריפי Opus:** כשבקשה נוגעת בסייבר, ביולוגיה או כימיה, Fable מעביר אותה ל-Opus 4.8, והתשובה מסומנת. הדוברים מודים שחוקרים לגיטימיים ייחסמו לפעמים [14:15].

- **נתוני שימוש:** נפח ה-API עלה פי 17 כמעט בתוך שנה [6:28], ומפתח ממוצע מבלה 20 שעות בשבוע עם Claude Code [6:28].

- **Managed Agents מוצג כמכפיל מהירות:** לפי Angela, חברות שבנו עליו עשו זאת "10 times faster", כי לא נאלצו לבנות לבד harness, context ותשתית [23:26]. הדוגמאות: Notion ו-Asana.

- **פיצ'רים חדשים בפלטפורמה:** scheduled deployments, vaults למשתני סביבה עם credentials, ו-outcomes שבו agent חוזר על העבודה עד שהוא עומד ב-rubric. נוסף להם dreaming: agent שעובר על סשנים קודמים ומעדכן את ה-memory ואת ה-skills [6:28, 26:29, 28:02].

- **Claude Code עובר מ-agent יחיד לניהול צי:** Agents View ב-Claude Code, CLI, Desktop, routines, Claude Security ו-dynamic workflows שמריצים עשרות עד מאות agents במבנה דטרמיניסטי [32:37–35:41].

- **תוצאות מלקוחות:** Spotify ממזגת יותר מ-1,000 PRs בחודש וקיצרה את זמן המיגרציה ב-90%. ב-Mercari התפוקה ההנדסית עלתה ב-90% משנה לשנה. ב-Rakuten יש release גדול כל שבועיים במקום פעם ברבעון [37:12, 3:28].

- **המלצה למפתחים: לתכנן לקראת המודל הבא:** להשתמש ב-primitives פשוטים כמו file system ו-sandbox במקום harness מסובך, לכתוב evals קשים יותר ולהפוך את החלפת המודל לתהליך אוטומטי [17:18–18:52].

מה לא נאמר

- **אין מספרים ל-benchmarks:** נאמר ש-Fable 5 מוביל ב-SWE-Bench Pro וב-Frontier Bench של Cognition, אבל לא נמסר אף ציון.

- **אין תמחור:** לא נאמר כמה עולה Fable 5, כמה עולים Managed Agents או הפיצ'רים החדשים. נאמר רק שבקשות שמנותבות ל-Opus 4.8 מחויבות לפי תעריפי Opus.

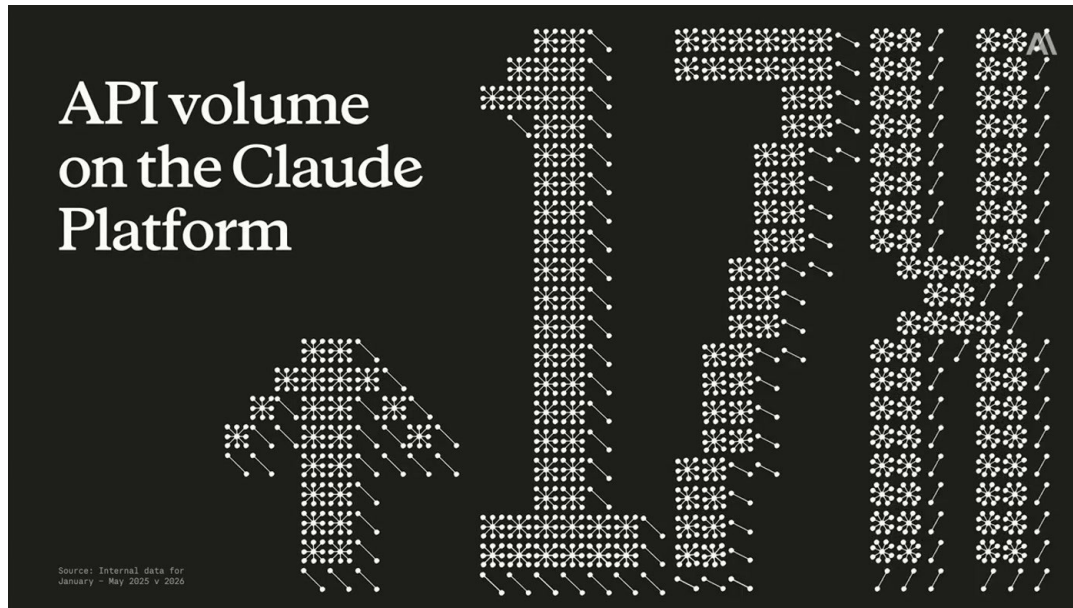
- **ה-safeguards לא נמדדו בפומבי:** אין נתון על שיעור החסימות השגויות. לא הוסבר איך מזהים שבקשה "נוגעת" בתחום רגיש ולא מה עושים במקרה של ניתוב שגוי.

- **המספרים של הלקוחות לא מגובים במתודולוגיה:** 8x קוד ב-Antropic, 90% ב-Mercari ו-10x ב-Managed Agents הם נתונים שנמסרו מהבמה בלי הגדרה של מה נמדד.

- **dreaming לא הוערך:** לא הוצגה שום ראיה שהוא באמת משפר את ביצועי ה-agent לאורך זמן.

- **vaults לא הודגם:** הוא הוזכר בפתיחה ומופיע כשדה בטופס ה-deployment, אבל לא הוצג בפועל.

2. פלטפורמה וחזון: פער האקספוננט

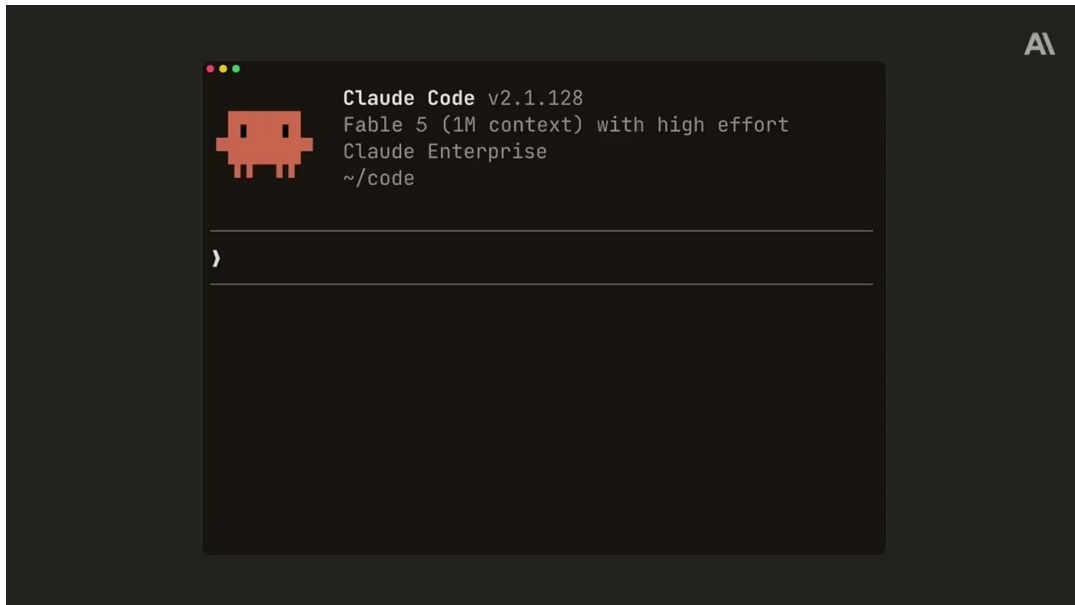


שקף "API volume on the Claude Platform": חץ עולה והמספר 17, הגידול בנפח ה-API בתוך שנה

Katelyn פותחת בהצגת Anthropic כחברת פלטפורמה: מפתחים יוצרים על גבי הפלטפורמה יותר ערך ממה ש-Anthropic הייתה יכולה לבנות לבד. ההמחשה הראשונה היא Rakuten. הם התחילו להשתמש ב-Claude Code כדי להאיץ פיתוח, ומשם עברו לבנות agents פנימיים על Managed Agents בתחומי engineering, מוצר, מכירות וכספים. מנהל מוצר אצלם "מנהל צוותים של agents" כמו שמנהלים צוותים של אנשים, והתוצאה היא release גדול כל שבועיים במקום פעם ברבעון [3:28]. הדוגמה השנייה היא Canva Code, שמאפשר למשתמשים שלא כותבים קוד לבקש בתוך עיצוב mini-app אינטראקטיבי, כמו מפה, מחשבון או widget.

אחר כך מגיע ציר הזמן: לפני שנתיים המודל ידע לנסח commit message. לפני שנה Opus 4 בנה פיצ'ר שלם. לפני חצי שנה agents רצו לילה שלם. לפני חודשיים Mythos קרא את כל קוד המקור של OpenBSD ומצא חולשה בת 27 שנים. ומכאן הטענה: "These jumps keep getting bigger, but the time intervals keep getting shorter." [4:58]. מול זה "most business capabilities are still on a linear" [4:58], והפער ביניהם הוא ההזדמנות. הנתונים שהוצגו: "Year over year, API volume is up nearly 17 times on the platform." [6:28]. הוכרזו גם שני פיצ'רים ב-Managed Agents: הרצה לפי לוח זמנים (scheduled deployments) ו-vaults לשמירת משתני סביבה, כך ש-agents יכולים לבצע קריאות API מאומתות בלי לקבל גישה למפתחות עצמם [6:28].

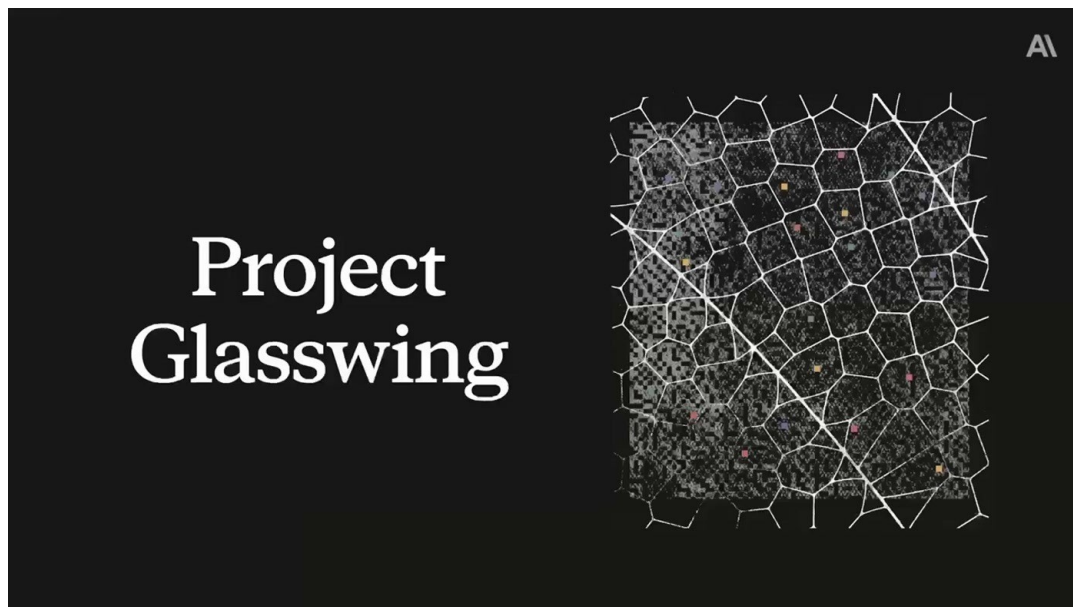
3. המודלים: Fable 5 ו-Mythos 5



מסך Claude Code v2.1.128 עם "Fable 5 (1M context) with high effort" בחשבון Claude Enterprise: המודל החדש כבר מופיע בבחירת המודל של הכלי

Diane, שליוותה לדבריה 21 גרסאות של Claude, מסבירה למה Fable 5 משמעותי, ומתחילה בקוד. ה-benchmark, לדבריה, "undersells the story": ככל שהמשימה ארוכה ומורכבת יותר, הפער ממודלים אחרים גדל. היא מצביעה על שני גורמים. הראשון הוא single-shot correctness: בעיה מורכבת ומוגדרת היטב נפתרת כבר בניסיון הראשון, ובודקים מוקדמים סיפרו על prompt יחיד שהפיק עבודה של ימים או שבועות. השני הוא long horizon autonomy: "Fable 5 can run for days on a single goal and stay coherent the entire way." [11:07]. לדבריה המודל זוכר את המפרט גם במשימות שמשתרעות על מיליוני tokens, ומנהל sub-agents באמינות טובה יותר ובחיסכון רב יותר מכל מודל קודם.

עוד נקודה שהיא מדגישה: Fable 5 טוב עוד יותר בקריאת קוד מאשר בכתיבתו. הוא מתחקר תקלות טוב יותר, חופר בהיסטוריה של ה-repo כדי למצוא מה נשבר ומתי, ומציע שיפורים ביוזמתו. מחוץ לקוד הוא מטפל מקצה לקצה בניתוח פיננסי, מסמכים, מצגות וגיליונות, מתמודד טוב יותר עם בקשות מבולגנות, ולדבריה יכולות הראייה שלו (vision) הן הטובות בתעשייה בקריאת דיאגרמות, גרפים ומסכי אפליקציה. לקוחות: Rakuten מציינת שבדרגות ה-effort הגבוהות המודל בודק את העבודה של עצמו. Cognition דיווחה שהוא קיבל את הציון הגבוה ביותר ב-Frontier Bench שלה. GenSpark דיווחה שהוא ניצח את כל המודלים שבדקה, בעיקר בעיצוב UI ובפיתוח משחקים [15:47–14:15].



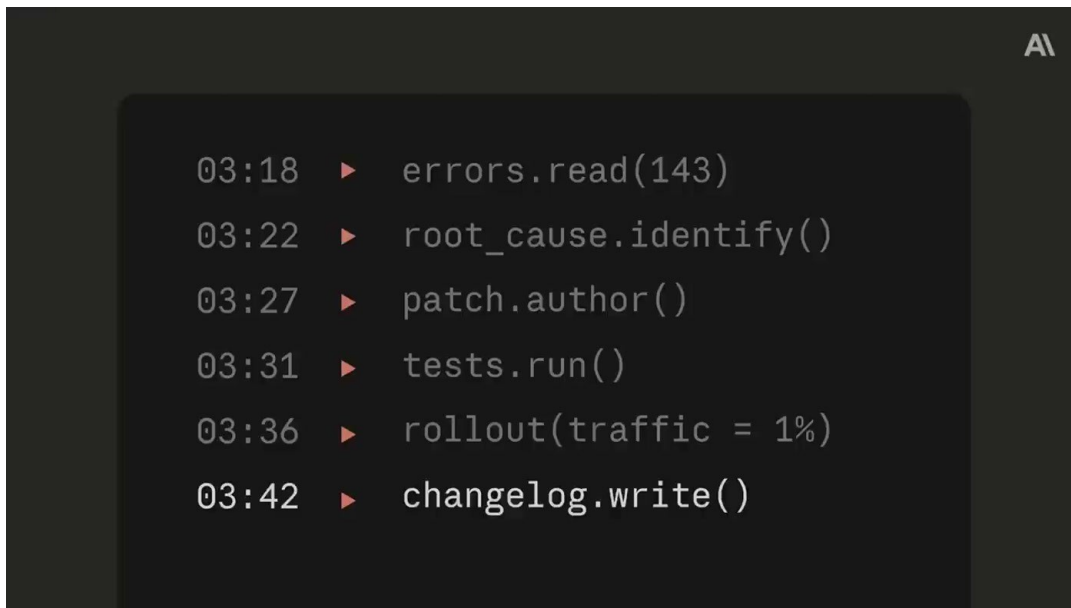
שקף "Project Glasswing": התוכנית שבמסגרתה Mythos Preview ועכשיו Mythos 5 זמינים רק לקבוצת שותפים מצומצמת

Diane מודה שליכולת הזו יש גם צד מסוכן. לפני חודשיים Mythos Preview נפתח רק לשותפי Project Glasswing, כי יכולות הסייבר שלו היו חזקות מספיק כדי שאפשר יהיה לנצל אותן לרעה. מאז נבנתה מערכת safeguards שמאפשרת להפיץ לכולם את אותה רמת אינטליגנציה דרך Mythos 5. Fable 5. Mythos הוא אותו מודל, "but with the cyber and bio safeguards lifted" [15:47], והוא זמין לשותפי Glasswing. בהמשך החודש תיפתח הרשמה לחוקרים ממדעי החיים, כי לטענתה אותן יכולות שהופכות ביולוגיה לתחום מסוכן הן גם אלה שיכולות להביא את התועלת הגדולה ביותר.

Diane Penn [14:15] When a request first touches cybersecurity, biology, or chemistry topics, Fable — will route it to our next most capable model, Opus 4.8. The response is clearly labeled, and you're charged Opus prices. We know this is not perfect yet

- **לתכנן לקראת המודל הבא:** First, you should design for the next version of Claude, not just the "current one." [17:18]. לדבריה, מי שמנצח הם צוותים שה-architecture וה-harness שלהם מוכנים לקפיצה הבאה.
- **primitives פשוטים במקום harness מתוחכם:** ככל שהמודלים חכמים יותר, הם משיגים יותר עם file system ו-sandbox ופחות עם harness מורכב מדי [17:18].
- **evals קשים יותר ואבות טיפוס שעוד לא עובדים:** כשאב טיפוס שנכשל מתחיל פתאום לעבוד, זה הסימן שהיכולות התקדמו ושהגיע הזמן לשחרר משהו חדש [18:52].
- **להפוך את החלפת המודל לקלה:** evals ותהליכי בדיקה אוטומטיים, ובדיקה ידנית שוטפת של כל גרסה חדשה. לדבריה, שדרוג מודל הוא הזדמנות עסקית [18:52].
- **time horizon כמדד:** כמה זמן מודל עובד לבד עד שהוא מאבד את הכיוון. הדוגמה שלה: במקום לבקש עדכון סטטוס לפרויקט, מבקשים מ-Fable לדאוג שהפרויקט יישאר במסלול לאורך כל השבוע [17:18–15:47].

Claude Managed Agents: harness, context, .4 infrastructure

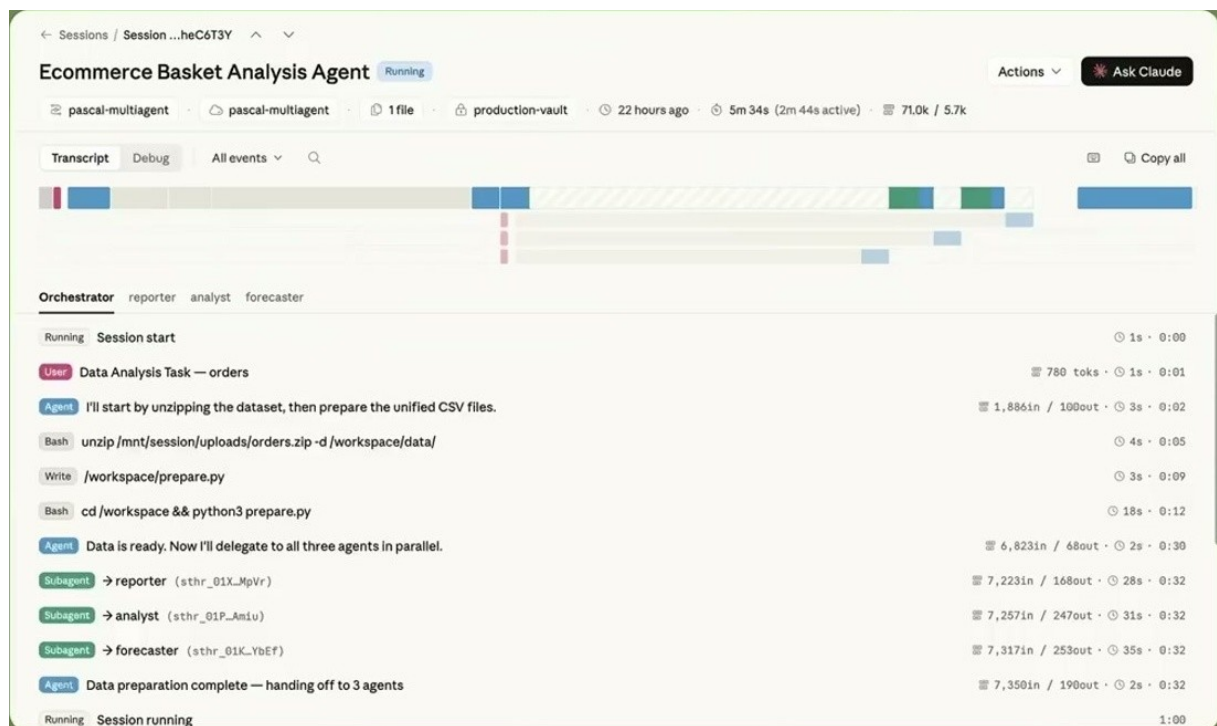


```
03:18 ▶ errors.read(143)
03:22 ▶ root_cause.identify()
03:27 ▶ patch.author()
03:31 ▶ tests.run()
03:36 ▶ rollout(traffic = 1%)
03:42 ▶ changelog.write()
```

ציר זמן של תיקון אוטומי: $errors.read(143) \leftarrow root_cause.identify() \leftarrow patch.author() \leftarrow tests.run() \leftarrow rollout(traffic = 1\%) \leftarrow changelog.write()$, בתוך 24 שניות

Angela פותחת בתרחיש: בלילה מוצר זיהה שהוא שבור, קרא את דוחות השגיאה שלו, מצא את הבאג, כתב תיקון והפיץ אותו, וכשהצוות התעורר גם ה-changelog כבר היה כתוב. בלי stand-up ובלי ticket. בעל פה נאמר שהתיקון הופץ "to all the users", אבל בשקף מופיע שלב ביניים של rollout ל-1% מהתעבורה, פרט שלא הוזכר בהרצאה. לדבריה, זו חברה שהעבודה עצמה רצה בה על AI, ואנשים רק מחליטים מה התוצאה הרצויה. כדי להגיע לשם צריך שלושה מרכיבים.

Harness: כלים, סביבה והרשאה לפעול. ב-Managed Agents ה-harness מפריד בין ה"מוח" ל"ידיים": "The brain decides what to do, and then we spin up sandboxes or hands to execute the work as needed." [21:54]. כמו כן מגדירים outcome, וה-agent חוזר על העבודה עד שהוא מגיע אליו. Context: חלון של מיליון tokens בלי ירידה באיכות, memory, יכולת לקרוא ולכתוב skills כדי להשלים פערי ידע, ו-dreaming. Infrastructure: הקמה וסגירה אוטומטית של sandboxes והפעלת צי של agents לפי הצורך. Notion משתמשת בזה ל-orchestration של agents בתוך המוצר, ו-Asana בנתה "AI teammates" שעובדים לצד אנשים בתוך הפרויקטים [24:57–23:26].

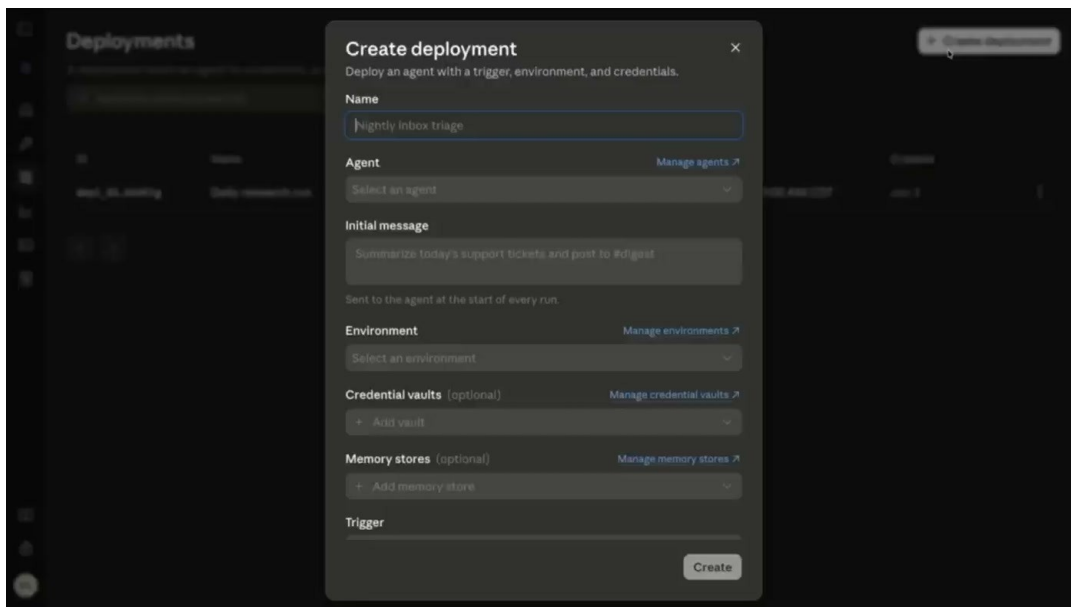


תצוגת טקסט ב-Claude Developer Console עבור "Ecommerce Basket Analysis Agent", תמונה, קריאות לכלים והעברת משימות ל-sub-agents

Katelyn מדגימה את המוצר דרך שותפות Claude עם Atlassian Williams F1 וקבוצת מרוצים בדיונית, Shinkiro Racing. בדשבורד יש ארבעה פרויקטי מחקר: אווירודינמיקה, טמפרטורת צמיגים, יחידת הכוח ובטיחות הנהג. כל אחד מהם מגובה ב-Managed Agent נפרד. הפרויקטים משתמשים ב-rubric: outcomes מגדיר מה נחשב תוצאה טובה, וה-agent חוזר על העבודה עד שהוא עומד בו. שאלה בשפה חופשית (כמה שוקל הרכב ואיך המשקל מתחלק) מרעננת את הדשבורד, והתשובה היא 800 ק"ג שמתחלקים בין יחידת הכוח, הסוללה ושאר הרכיבים [26:29].

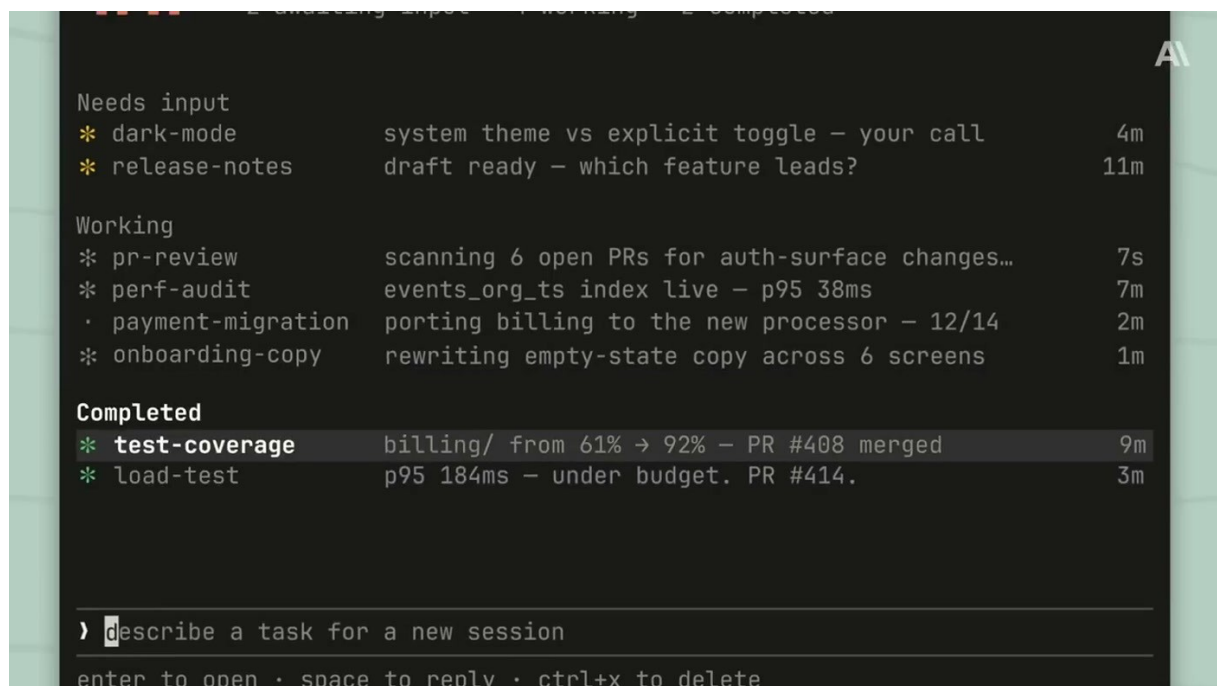
בצד המפתח, ה-Console הוא מרכז העבודה: היסטוריית ששנים, תשובות וקריאות לכלים. מהיום אפשר ליצור Deployment שרץ לפי לוח זמנים, למשל בדיקת בטיחות כל בוקר בתשע. memory מאפשר ל-agent לכתוב תובנות ל-file system בזמן העבודה. dreaming מריץ agent חדש שעובר על כל הסשנים הקודמים ומעדכן את ה-memory ואת ה-skills [28:02]. לדברי Katelyn, outcomes, scheduled deployments, ו-dreaming כבר זמינים.

— And lastly, we give our agents the ability to dream, so they can inspect over their own previous trajectories and identify how to self-improve [23:26] Angela Jiang



דיאלוג Create deployment: שדות Name, Agent, Initial message, Environment, Credential vaults, Memory stores, Trigger (אופציונלי) (אופציונלי).

5. Claude Code: מ-agent יחיד לצי של agents



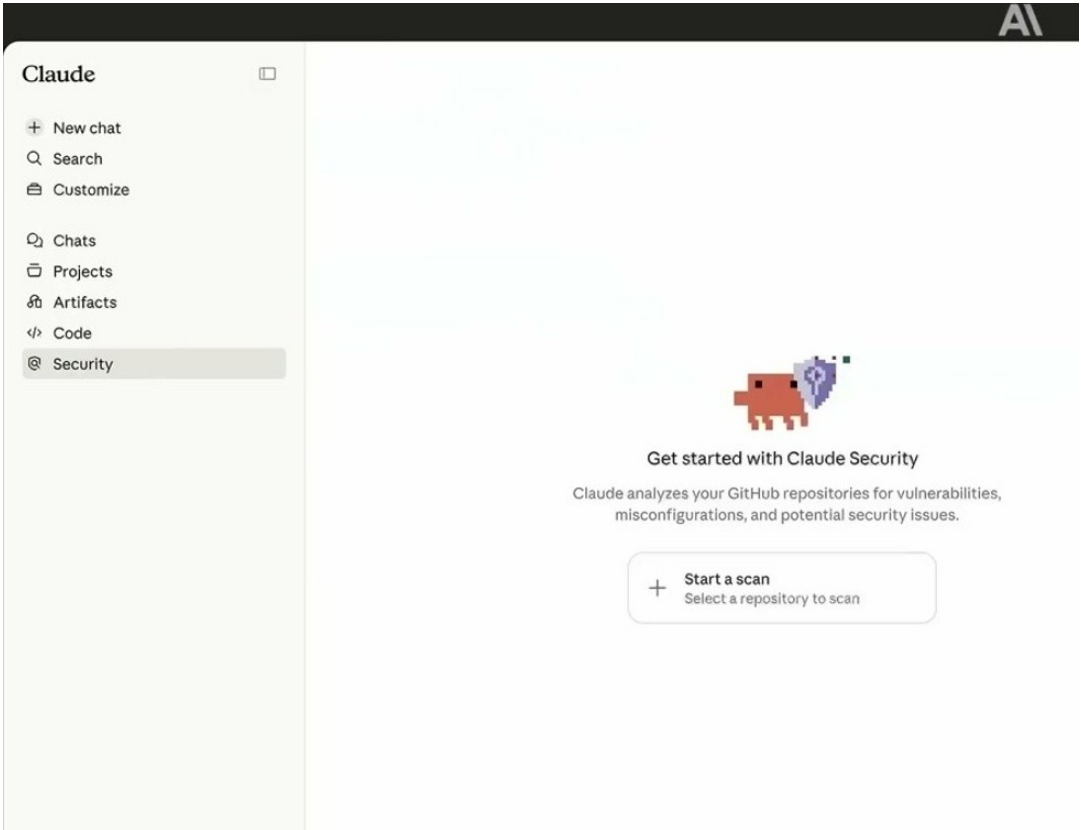
Claude Agents View ב-CLI: שנים לפי סטטוס. Needs input (dark-mode, release-notes), Working (pr-review, perf-audit, payment-migration, onboarding-copy) Completed (test-coverage, load-test, PR #408 merged)

Kat מגדירה את המשימה של Claude Code כגישור בין רעיון למוצר שעלה לאוויר, ומשווה את הצוות למטפסי הרים שמתקדמים יחד עם המשתמשים בשטח שעוד לא מופה. השינוי שהיא מתארת: לפני שנה היא בדקה כל edit בנפרד. היום רבים עובדים ב-auto mode ובדקים רק PR שכבר נבדק. מזה נולדו ממשקים חדשים. ה-CLI נשאר למשתמשים מתקדמים, ה-IDE מיועד למי שרוצה לעקוב אחרי השינויים, ובשביל מי שמריץ כמה סשנים במקביל ("multi-claude") נוספו שני ממשקים: Claude Code ב-Claude Desktop, ממשק גרפי עם previews ו-sidebar שמרכז סשנים מקומיים וסשנים בענן, ו-Agents View, לוח בקרה שנשאר בתוך הטרמינל [32:37]. ה-IDE extension ואפליקציית ה-Desktop בנויים על Claude Agent SDK.

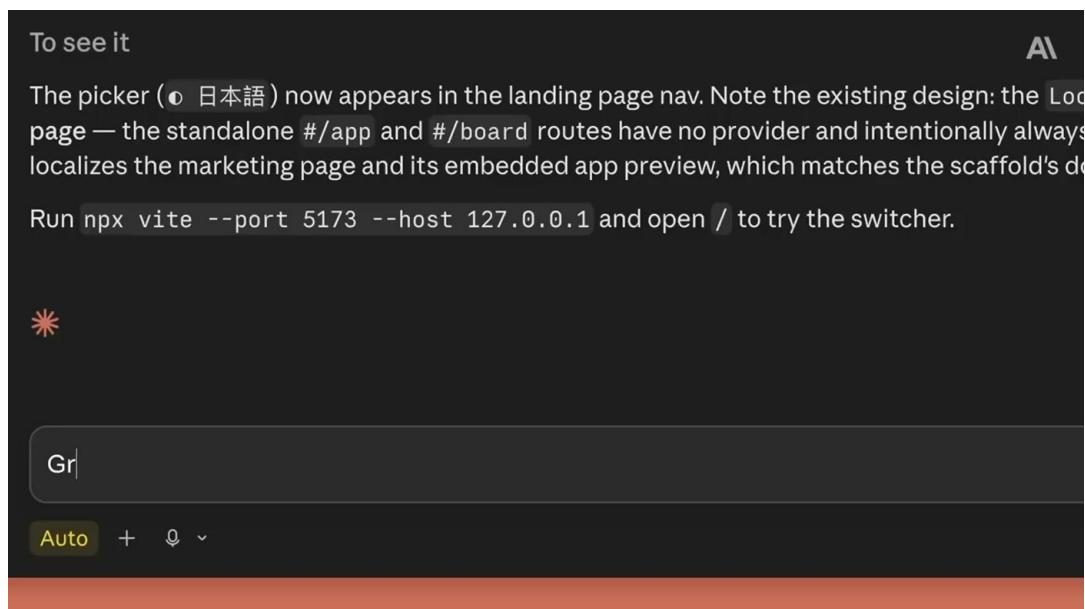
הנתון שהיא מביאה מ-Antropic: "At Anthropic, engineers on average ship 8x more code than they did in past years," [32:37], וזאת אף שצוות ה-engineering גדל משמעותית. מלקוחות: Spotify בנתה על

ה-Agent SDK agent שרץ ברקע, קורא תוכנית מיגרציה באנגלית פשוטה ומפעיל צי של agents שפותרים PRs. "They're merging over 1,000 PRs a month into production, and they've cut migration time down by 90%." [37:12] ב-Mercari כל צוות ה-engineering עובד עם Claude Code, והתפוקה עלתה ב-90% משנה לשנה.

מה המשתמשים ביקשו	מה נבנה	מה זה עושה
פחות זמן על code review	Code Review	צוות agents שמחפש באגים קריטיים ב-PRs. לדבריה, אלפי חברות משתמשות בו מדי יום, כולל כל הצוותים ב-Anthropic
לקודד מכל מקום	Remote Control + אפליקציות iOS/Android	שליטה בסשנים מהטלפון
להריץ על tickets חדשים	Routines	מגדירים פעם אחת, והם מופעלים לפי לוח זמנים, webhook או קריאת API
צוותי האבטחה לא עומדים בקצב	Claude Security	סריקה לילית של ה-codebase, סימון חולשות לפי חומרה ופתיחת טשן לתיקון
משימות ענק: refactors ומיגרציות	Dynamic workflows	עשרות עד מאות agents שרצים במקביל במבנה דטרמיניסטי



מסך הפתיחה של "Claude Security: "Get started with Claude Security" וכפתור לבחירת repository לסריקה



סיכום הסשן בדמו הלוקליזציה: בורר שפות (日本語) נוסף לניווט בדף הנחיתה, והוראה להריץ `npx vite --port 5173` כדי לבדוק אותו

הדמו של Kaizen Operations: אתר שיווקי באנגלית צריך תרגום ל-13 שווקים. סשן יחיד של Claude Code It took Claude about three minutes to complete this one" בנה בורר שפות ותרגם ליפנית. [38:44] translation, and we have 12 more to go." בהרצה סדרתית זה היה לוקח כמעט שעה ודורש הרבה prompts ידיניים. במקום זה הם ביקשו מ-Claude ליצור 12 agents dynamic workflow. תרגמו במקביל, ואחריהם הופעלו 12 agents נוספים לבדיקת התוצאות, והכול הוצג בחלונות הצד של אפליקציית ה-Desktop. את ה-workflow אפשר לשמור כקוד JavaScript ולהשתמש בו שוב. לדבריה, אותו מנגנון מתאים למיגרציות גדולות, ל-audit של codebase ולאופטימיזציות ביצועים. Fable 5 זמין לכל משתמשי Claude Code [40:16].

6. מה עושים עם זה

- **לבדוק את Fable 5 על המשימות הארוכות שלכם:** לפי הדוברים, היתרון שלו בא לידי ביטוי בעיקר במשימות ארוכות ומורכבות, לא ב-benchmark הסטנדרטי. זה המקום שבו כדאי למדוד.
- **להיערך לניתוב של ה-safeguards:** אם המוצר נוגע בסייבר, ביולוגיה או כימיה, כדאי לבדוק כמה בקשות מנותבות ל-Opus 4.8, איך הניתוב משפיע על העלות ואיך מזוהים בתשובה שהיא מסומנת כמנותבת.
- **להשקיע ב-evals אוטומטיים לפני השדרוג הבא:** ההמלצה החוזרת בהרצאה: שדרוג מודל צריך להיות פעולה שגרתית ולא פרויקט.
- **לבחון את Managed Agents מול ה-harness הפנימי:** להשוות את outcomes, memory, dreaming ו-vaults ל-harness שבניתם בעצמכם, ובעיקר את העלות של תחזוקת התשתית.
- **לנסות dynamic workflows על משימה חוזרת:** לוקליזציה, מיגרציה או audit. ה-workflow נשמר כ-JavaScript, ולכן אפשר לשמור אותו ב-version control.
- **לשקול routines ו-Claude Security ב-CI:** שני הכלים מעבירים לאוטומציה עבודה שהיום מישהו מתחיל ידנית.

שאלה פתוחה	למה זה חשוב
מה המחיר של Fable 5 ושל Managed Agents?	בלי זה אי אפשר לחשב ROI, במיוחד ב-agents שרצים ימים שלמים
כמה חסימות שגויות יש ב-safeguards?	הדוברים הודו שחוקרים לגיטימיים ייחסמו, אבל לא נתנו נתון
האם dreaming באמת משפר ביצועים, ואיך מונעים ממנו להצטבר כ-skills שגויים?	לא הוצגה הערכה, ועדכון עצמי של memory ו-skills הוא סיכון
מה עומד מאחורי ה-8x וה-90%?	לא הוסבר מה נמדד (שורות קוד? PRs?) ומול איזה בסיס השוואה
איך vaults מנהלים credentials בפועל?	הפיצ'ר הוכרז ומופיע בטופס, אבל לא הודגם

7. מילון מונחים

מונח	משמעות
Harness	השכבה שנותנת למודל כלים, סביבת עבודה והרשאה לפעול
Brain / hands	ההפרדה ב-Managed Agents בין המודל שמחליט מה לעשות לבין ה-sandboxes שמבצעים את העבודה
Outcomes / rubric	הגדרה של מה נחשב תוצאה טובה. ה-agent חוזר על העבודה עד שהוא עומד בה
Memory	קבצים שה-agent כותב בזמן העבודה וקורא בהרצות הבאות
Dreaming	agent שעובר על סשנים קודמים ומעדכן את ה-memory ואת ה-skills
Skills	ידע ויכולות ש-agent יכול לקרוא ולכתוב כדי להשלים פערים
Scheduled deployment	הרצת agent לפי לוח זמנים קבוע
Vault	אחסון מאובטח של משתני סביבה, כך שה-agent מבצע קריאות מאומתות בלי לראות את המפתח
Time horizon	כמה זמן מודל עובד לבד לפני שהוא מאבד את הכיוון
Single-shot correctness	פתרון נכון כבר בניסיון הראשון
Long horizon autonomy	היכולת לעבוד ימים שלמים על מטרה אחת ולשמור על כיוון
Project Glasswing	תוכנית גישה מוגבלת ל-Mythos לשותפים נבחרים
Auto mode	מצב ב-Claude Code שבו מעבירים ל-Claude הרשאות כדי שיעבוד בלי אישור לכל צעד
Multi-clauding	הרצה של כמה סשנים של Claude Code במקביל
Agents View	לוח בקרה בטרמינל לסשנים שממתנים לקלט, רצים או הסתיימו
Routines	הפעלה אוטומטית של Claude Code לפי לוח זמנים, webhook או קריאת API
Dynamic workflows	הרצה מקבילה של agents רבים במבנה דטרמיניסטי, שאפשר לשמור כ-JavaScript
Claude Agent SDK	ה-SDK שעליו בנויים ה-IDE extension ואפליקציית ה-Desktop, וגם agents של לקוחות