Claude Code נראה צודק, אבל פספס 800 אלף דולר

מה קרה כשהפעילו Claude Code על ניתוח רכש אמיתי אחרי מיזוג, ולמה הפתרון היה להכין את הנתונים במבנה מסודר עוד לפני שהסוכן נוגע בהם

Natalia Connolly, Kevin Madura (AlixPartners) · AI Engineer · 8 בנובמבר 2026
משך: כ-19 דקות · YouTube
סיכום אוטומטי — AI Briefing (aibriefing.dev)

על המסמך הזה

זו הרצאה קצרה (כ-19 דקות) מכנס AI Engineer. לפי השקפים היא הועברה ב-Leadership Session בניו יורק ב-14 באוקטובר 2026. מרצים בה Natalia Connolly מצוות AI and Data ו-Kevin Madura מצוות AI engineering, שניהם מ-AlixPartners, חברת ייעוץ שמתמחה בפשיטות רגל ובהבראת חברות, בייעוץ עסקי וטכנולוגי ובסיכונים וציות. ההרצאה מתארת פרויקט אמיתי: ניתוח רכש אחרי מיזוג של שתי חברות בריאות, עם מאות ספקים ויותר מ-40,000 מסמכי חוזה. בגלל סודיות הלקוח, הדוגמה שמוצגת היא שחזור סינתטי.

הטענה המרכזית: כש-Claude Code מקבל נתוני חוזים גולמיים, הוא מחזיר תשובות שנראות שלמות ומשכנעות, אבל חלק מהן שגויות, ואף אחד לא מתריע על כך. במקרה שלהם שווי החוזים הכולל היה שגוי בכ-800 אלף דולר, ותאריך ההודעה שנדרש כדי למנוע חידוש אוטומטי איחר בחודש. הפתרון שהם מציגים: להשקיע מראש ב-pipeline שמארגן את הנתונים במבנה קבוע (טבלאות מוגדרות מראש), לבנות כלים מותאמים שהחוקים העסקיים כתובים בתוכם, ולתת לסוכן את ההקשר העסקי. בדרך הם מונים חמישה לקחים מהשטח.

איך לקרוא את המסמך

- **1. תקציר מנהלים:** הטענות המרכזיות עם חותמות זמן, וגם מה לא נאמר.
- **2. ההקשר:** העסקה, השאלות של צוות הרכש, ושתיים מחמש התשובות של הסוכן שהיו שגויות.
- **3. למה נתוני חוזים מבולגנים:** ארבעה מקרים שבהם נתונים גולמיים מטעים את הסוכן.
- **4. Pipeline בחמישה שלבים:** שבועיים של בנייה, עיבוד של כל החוזים בלילה אחד, ו-DSPy לחילוץ נתונים מתרשימים.
- **5. כלים מותאמים:** איך החוקים העסקיים נכנסים לתוך הכלים ולא נשארים רק ב-prompt.
- **6. חמשת הלקחים:** תאריך שגוי, תיקונים לחוזים שלא קושרו לחוזה המקורי, שגיאות גישה שנראות כמו אפס חיסכון שנספר פעמיים, והקשר עסקי.
- **7. מה לעשות עכשיו:** צעדים מעשיים וטבלת שאלות פתוחות.
- **8. מילון מונחים.**

הערת מקור המסמך נוצר אוטומטית מהתמליל ומהשקפים של ההקלטה. מערכת זיהוי הדיבור (ASR) משבשת לפעמים שמות ומונחים. למשל, בתמליל מופיעים "Alex Partners" במקום "AlixPartners, "cloud code" במקום Claude Code ו-"dspi" במקום DSPy. במקומות כאלה תיקנו לפי ההקשר וסימנו כשלא היינו בטוחים. הציטוטים מועתקים מילה במילה מהתמליל ונבדקו מול חותמת הזמן שלהם. לא נוסף מידע ממקורות חיצוניים.

1. תקציר מנהלים

זה לא סיפור על מודל חלש. הבעיות נבעו מנתונים שלא סודרו מראש ומחוסר בהקשר עסקי. הסוכן עשה מה שהתבקש וחישב נכון על מה שקיבל, אבל חלק מהנתונים חסר, חלק הופיע פעמיים, וחלק היה בלי הסבר מה הוא מייצג. המרצים מציגים שתי דרכים מעשיות לטפל בזה: לארגן את הנתונים במבנה קבוע מראש, ולכתוב את החוקים העסקיים בתוך הכלים.

- **תשובות משכנעות ושגויות:** מתוך חמש תשובות, שלוש היו נכונות: ההוצאה השנתית, החלוקה בין החברות ומספר החוזים. שתיים היו שגויות: שווי החוזים הכולל סטה בכ-800 אלף דולר, ותאריך ההודעה יצא 1 בדצמבר במקום 1 בנובמבר.
- **800 אלף הדולר נבעו מתיקון אחד שלא קושר לחוזה המקורי:** ה-amendment לא היה מקושר לחוזה האב, ולכן Claude Code לא מצא אותו.
- **סכימת סכומים בלי לקרוא את הטקסט טועה:** כשכל amendment מחליף את השווי הקודם במלואו, סכימה מגיעה לכ-17 מיליון דולר. התשובה הנכונה היא 7 מיליון, השווי בתיקון האחרון.
- **שבועיים של pipeline ולילה אחד של עיבוד:** pipeline בחמישה שלבים: עיבוד ראשוני ו-OCR, סיווג, חילוץ תנאים, תעריפים והוצאות, מיפוי לקטגוריות הוצאה, ורק אחר כך Claude Code. יותר מ-40,000 חוזים עובדו בלילה אחד.
- **מבנה קבוע מאפשר גם בדיקה:** הטבלאות המובנות עזרו לסוכן, וגם אפשרו לצוות לבדוק כל תשובה בקוד.
- **החוקים העסקיים כתובים בכלים, לא רק ב-prompt:** כלים כמו get rate card page או get specific vendor spend, ו-views במסד הנתונים, אוכפים את החוקים בקוד. בהמשך הם נארזים לשרתי MCP.
- **שגיאת גישה שנראית כמו אפס היא באג מסוכן:** כשהגישה לנתוני אחת החברות נכשלה, הסוכן הבין שההוצאה היא אפס. לפי השקף, הסוכן קבע שההוצאה של Summit היא 0$.
- **סוכנים שרצים במקביל סופרים את אותו חיסכון פעמיים:** נדרש שלב התאמה ואיחוד מוגדר מראש, שבנוי על DSPy.
- **הקשר עסקי עדיין מכריע:** המלצות נפסלו כי ההנהלה כבר טיפלה בהן שלושה חודשים קודם.

מה לא נאמר

- **אין שיעור שגיאות כולל:** מוצגות רק שתי שגיאות בתרחיש הסינתטי. לא נמסר כמה מהתשובות בפרויקט האמיתי היו שגויות לפני ה-pipeline ואחריו.
- **אין עלויות:** נאמר שכדאי להשקיע tokens וכסף מראש, אבל לא ניתנו מספרים על עלות העיבוד או צריכת ה-tokens.
- **אין השוואה לגישה הנאיבית בהיקף מלא:** הטענה שעבודה ידנית הייתה דורשת צוות גדול יותר היא הערכה בלבד ("I imagine") ולא מדידה.
- **ההערכה שהבעיה תיעלם בעוד 6–12 חודשים לא מבוססת:** Kevin עצמו מסייג ואומר "we'll see".
- **אין פרטים על בניית הכלים בפועל:** לא הוצג קוד, לא הוצגה הגדרת ה-MCP, ולא הוסבר איך נאכפת חסימת הפעולות.

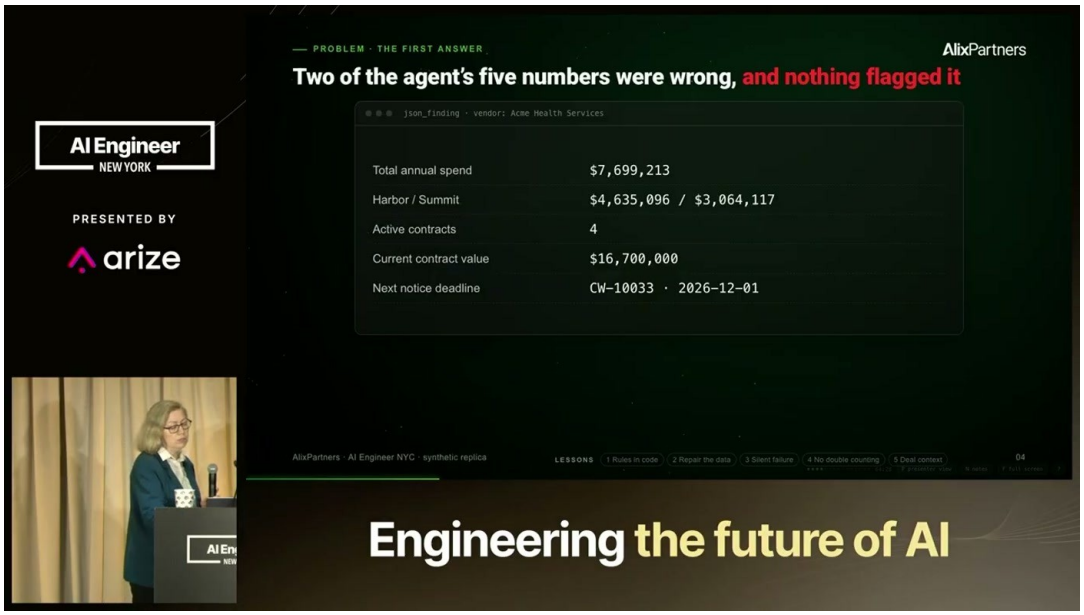
2. ההקשר: מיזוג, 40,000 חוזים ותשובות שנראו נכונות



תרחיש ההדגמה: Harbor Health רכשה את Summit Health Plan, ולשתיהן יש חוזים נפרדים עם אותו ספק, Acme. השאלה: איפה אפשר לחסוך ועד מתי צריך לפעול

חברת בריאות גדולה רכשה שחקן קטן ונישתי. אחרי המיזוג עמדו מול צוות הרכש מאות ספקים, יותר מ-40,000 מסמכי חוזה וכמה מסדי נתונים לא מקושרים, בלי תמונה כוללת של כל ספק. בתרחיש הסינתטי, Harbor Health (החברה הרוכשת) ו-Summit Health Plan (החברה הנרכשת) עובדות שתיהן עם Acme Health Services. צוות הרכש שואל שלוש שאלות: אפשר לקבל תנאים טובים יותר? כמה זה שווה? ועד מתי צריך לפעול?

הדרישה המרכזית היא שאפשר יהיה להתחקות אחרי כל תשובה עד המקור. מי שמנהל את המשא ומתן מול Acme צריך לענות על כל שאלה או התנגדות "with a specific citation, with a specific contract page or a specific invoice line".

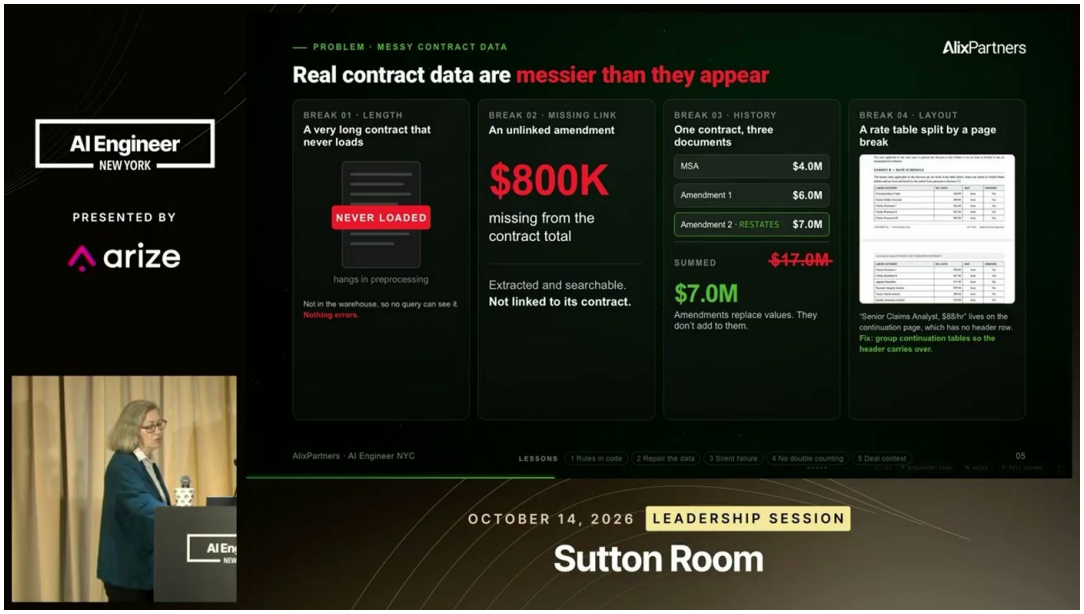


"Two of the agent's five numbers were wrong, and nothing flagged it": חמש התשובות של הסוכן, כולל ההוצאה של Harbor מול Summit, שווי החוזים ומועד ההודעה הקרוב (2026-12-01)

החזיר Claude Code תשובות שנראו מצוין: "They look complete. They're certainly well formatted. They look plausible enough.". הצוות בדק כל תשובה מול ה-warehouse. ההוצאה השנתית, החלוקה בין החברות ומספר החוזים היו נכונים. שווי החוזים הכולל היה שגוי בכ-800 אלף דולר. תאריך ההודעה למניעת חידוש אוטומטי

היה שגוי: "The agent said December 1st. The actual date was November 1st, right?". אילו סמכו על הסוכן, החוזה היה מתחדש אוטומטית.

3. למה נתוני חוזים מבולגנים



"Real contract data are messier than they appear": ארבעה מקרי כשל. חוזה ארוך שלא נטען, amendment יתום שגרם לחוסר של $800K, תיקונים שמחליפים זה את זה (סכימה מול $7.0M הנכון), וטבלת תעריפים שמתפצלת בין עמודים

Natalia פותחת בקביעה ש-"contract data are a lot messier than they appear", ומונה ארבעה מקרים:

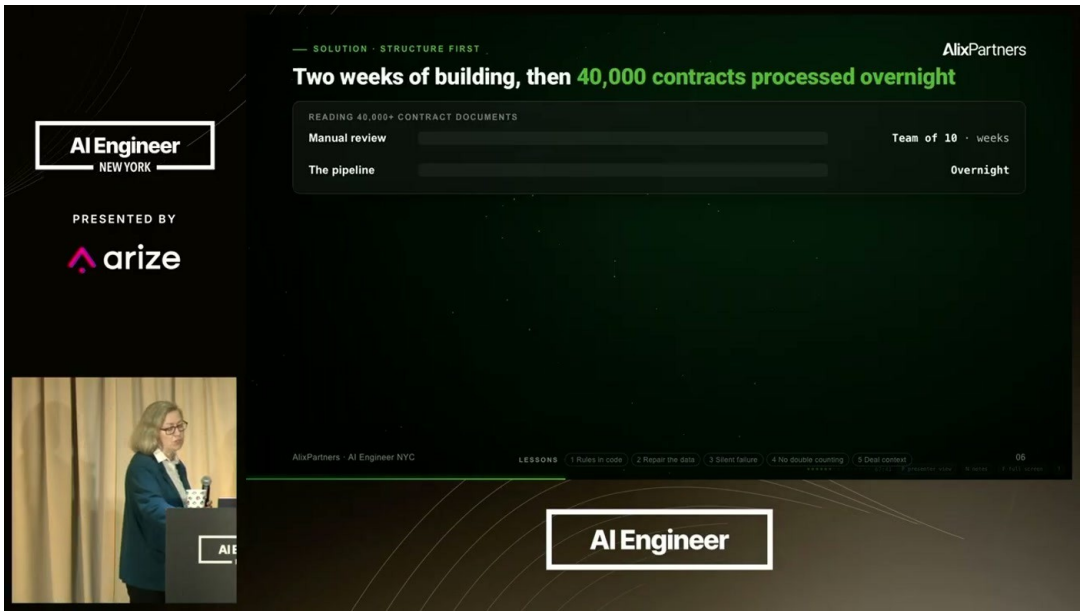
חוזים ארוכים שנתקעים בעיבוד הראשוני: השגיאה נתפסת כ-timeout, העיבוד ממשיך, והחוזה פשוט לא מגיע למסד הנתונים. אף אחד לא שם לב.

תיקונים יתומים: amendment שלא קושר לחוזה האב שלו. זו בדיוק הסיבה ש-Claude Code פספס את 800 אלף הדולר.

תיקונים שמחליפים את התנאים הקודמים: MSA ושני amendments, שכל אחד מהם מגדיר מחדש את השווי הכולל. מי שסוכם רק את הסכומים שחולצו מקבל כ-17 מיליון דולר. אבל הטקסט קובע שכל תיקון מחליף את השווי, ולכן "the correct answer is seven, which is the most recent amendment's value.".

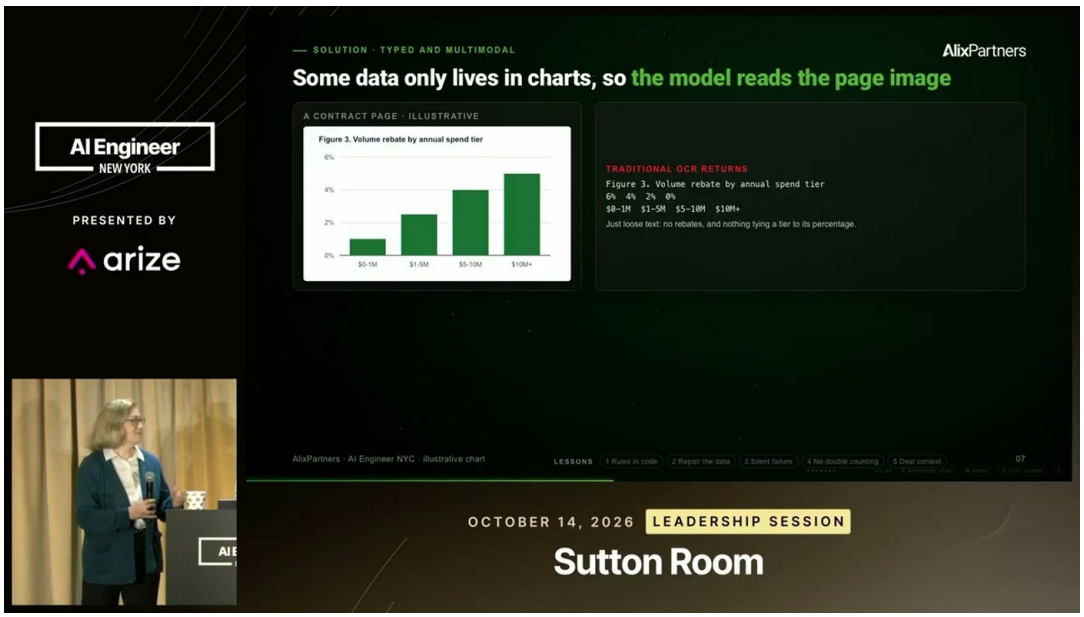
טבלאות ארוכות שעוברות בין עמודים: טבלת תעריפים (בתמליל "RAIDs", כנראה שיבוש של rates) שממשיכה לעמוד הבא בלי כותרת. הסוכן ינחש מה כל ערך מייצג. המסקנה של Natalia: "none of this is exotic".

4. Pipeline בחמישה שלבים ופלט מובנה



"Two weeks of building, then 40,000 contracts processed overnight": השוואה בין בדיקה ידנית ל-pipeline, עם זמני עיבוד לכל אחד

שלב	מה קורה בו
1. עיבוד ראשוני	Document intelligence ו-OCR
2. סיווג	קביעת סוג המסמך: MSA, amendment, NDA וכו'
3. חילוץ	תנאים, תעריפים והוצאות, כולל טבלאות שמתפצלות בין עמודים
4. מיפוי	שיוך כל שירות לקטגוריית הוצאה
5. ניתוח	רק בשלב הזה Claude Code עובד על הנתונים שחולצו

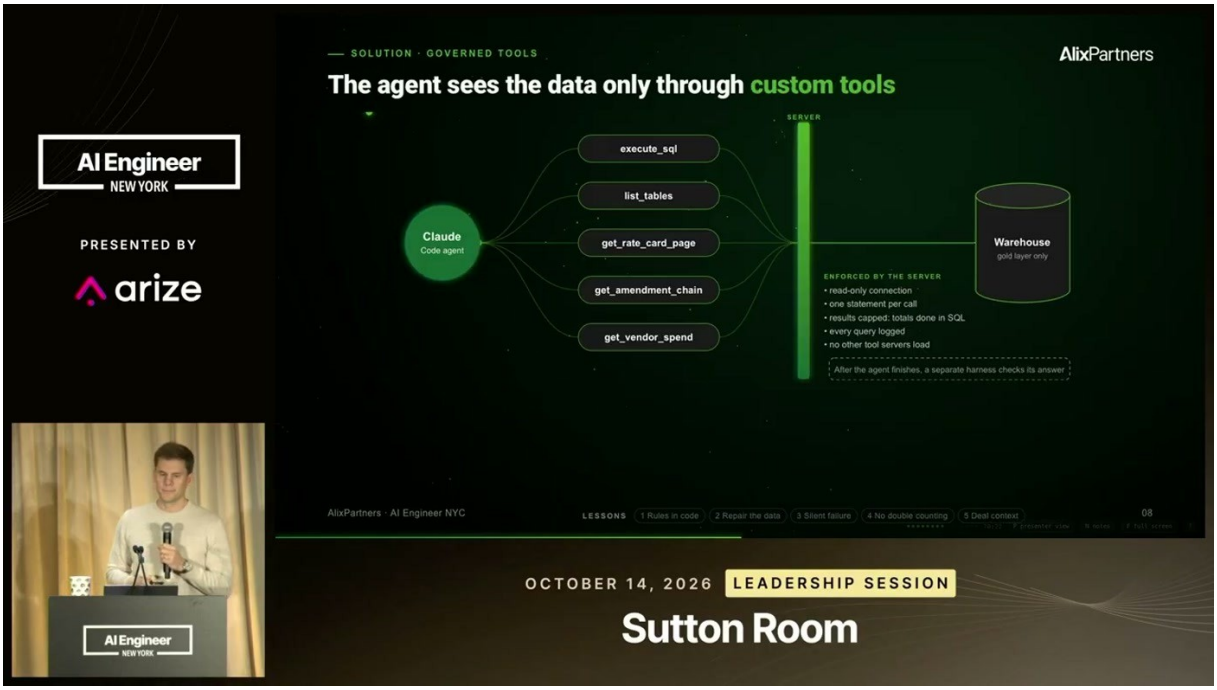


"Some data only lives in charts, so the model reads the page image": תרשים עמודות מתוך חוזה. משמאל מה ש-OCR נאיבי מחלץ, ומימין הפלט המובנה שמתקבל כשהמודל קורא את תמונת העמוד

לבניית ה-pipeline הקדישו "about two weeks", ואחר כך "40,000 plus contracts were processed overnight once it was done". לדברי Natalia, הטבלאות המובנות עזרו ל-Claude Code, וגם הקלו מאוד על בדיקה של כל תשובה בקוד.

Kevin מוסיף את הצד הוויזואלי. חלק מהמידע נמצא בתרשימים, ו-OCR נאיבי מחלץ את הצירים אבל מאבד את התוכן. הפתרון: להעביר את עמודי החוזה כתמונות דרך DSPy ולקבל פלט מובנה. הוא מסייג מראש: "this will likely not be relevant maybe like six to 12 months, we'll see". העיקרון הרחב יותר: "it's very tempting to just let Claude rip on the unstructured data and kind of have it figure it out, which is a valid approach". אבל כשיודעים מראש מה צריך, למשל מדרגות תעריף והנחות רטרואקטיביות (rebates), כדאי להגדיר את המבנה מראש ולתת ל-Claude למלא אותו.

5. כלים מותאמים עם חוקים עסקיים

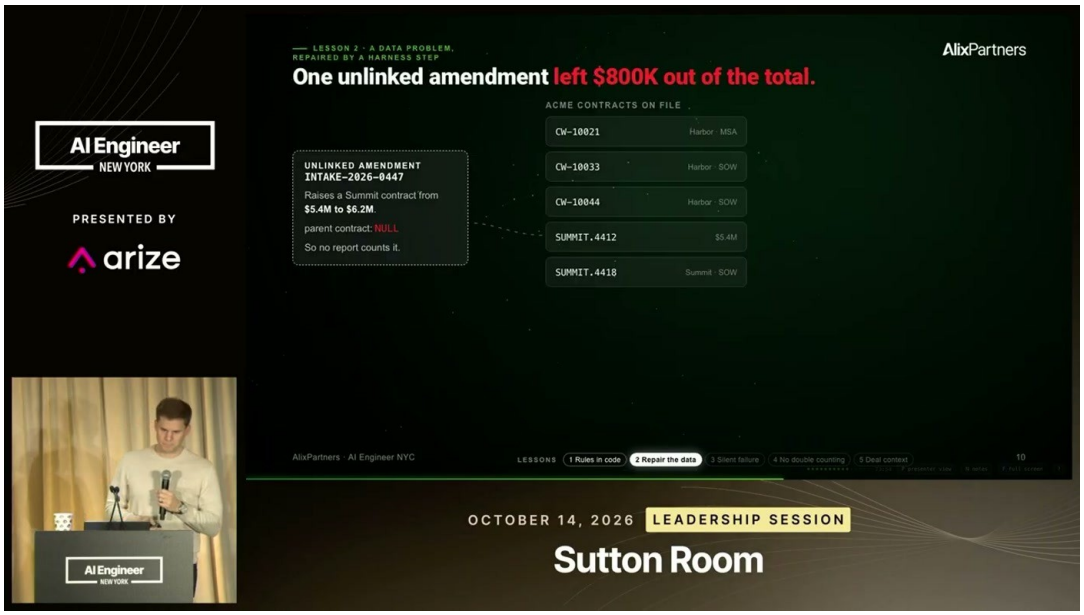


"The agent sees the data only through custom tools": Claude ניגש ל-Warehouse רק דרך כלים כמו execute_sql, list_tables, get_rate_card_page, get_amendment_chain ו-get_vendor_spend, כשהחוקים העסקיים כתובים בתוכם

אחרי שהנתונים מסודרים, Claude Code בונה לעצמו כלים. יש כלים כלליים כמו execute SQL ו-list tables, ויש כלים שמתאימים לעסקה הספציפית, כמו get rate card page או get specific vendor spend. בכלים האלה כותבים את החוקים העסקיים ואת המלכודות שלא רואים מהחוזה עצמו. כך גם סוכן חדש שמתחיל לעבוד על הנתונים מקבל קיצור דרך. Kevin מסביר למה Claude Code מתאים לזה: הוא רגיל לנווט לבד ב-codebases מבולגנים, וכאן זה אותו דבר, רק עם נתוני חוזים והוצאות.

כשמשהו עבד, ארזו אותו. לפעמים ל-prompt, אבל בעיקר לכלים ול-views במסד הנתונים, כדי שהחוקים ייאכפו בקוד ולא יהיו תלויים ב-prompt. משם ממשיכים לשרתי MCP והגדרות שנשמרות לאורך כל הפרויקט. כך הידע מצטבר. יתרון נוסף הוא בקרה על הגישה לנתונים. לדברי Kevin זה פשוט: "You just have a defined set of operations that you're comfortable with, and you block everything else".

6. חמשת הלקחים



"One unlinked amendment left $800K out of the total": רשימת החוזים מול Acme, ולצידה amendment שלא קושר לחוזה האב ולכן לא נכנס לסכום

לקח 1, התאריך השגוי: הסוכן לקח את התאריך מסעיף הסיום (termination), אבל השאלה דרשה תאריך שמחושב לפי סעיף החידוש (renewal). זה נראה פרט שולי, אבל לדברי Kevin "it's a lot of these nuanced issues which compound your error rates". מסקנה: לכתוב חוקים כאלה מראש.

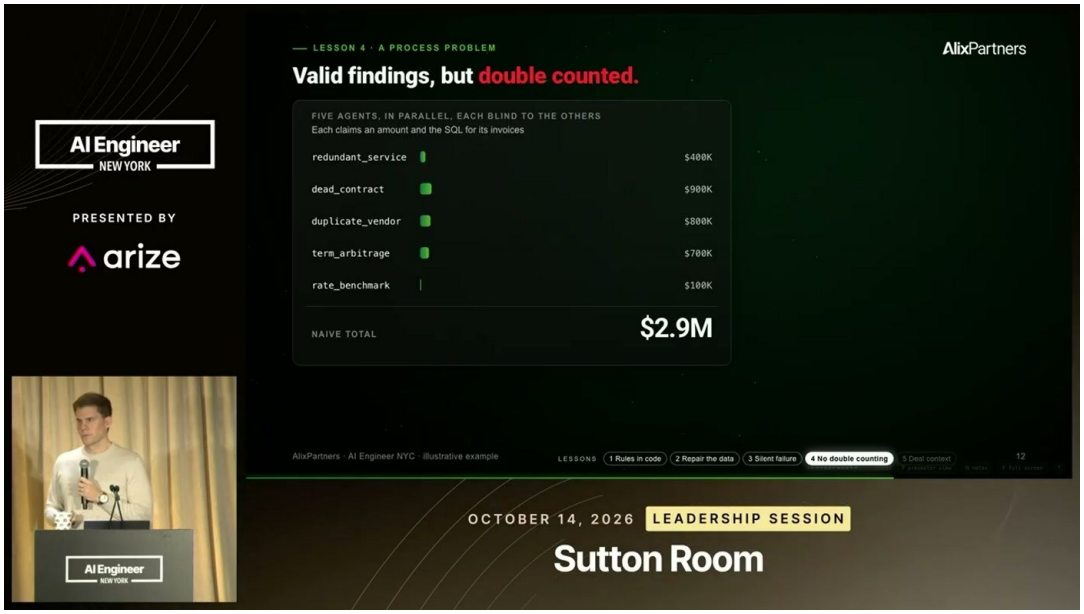
לקח 2, תיקונים יתומים: לארגונים רבים יש מערכת לניהול חוזים (CLM), ולגביהן Kevin אומר: "They're almost never maintained properly or well.". לכן משתמשים בנתונים מה-CLM, אבל גם נותנים ל-Claude לקרוא את תוכן החוזים ולבנות מחדש את ההיררכיה: MSA, ה-SOWs שתחתיו והתיקונים להם. כך אפשר לקשר מסמכים שלא קושרו, להימנע מספירה כפולה ולתפוס את המצב העדכני של היחסים עם הספק.



"Access controls hid Summit's spend, and the agent called it $0": ההוצאה מול Acme, Harbor $4,635,096 ו-Summit $3,064,117. בגלל בעיית הגישה הסוכן דיווח על $0 עבור Summit

לקח 3, שגיאת גישה שנראית כמו אפס: הגישה לנתוני שתי החברות התבצעה בדרכים שונות. בחלק מהמקרים חזר ערך פגום או אפס, או שגיאת הרשאה, וה-tool call לא החזיר ל-Claude מידע שמסביר מה קרה. הוא המשיך הלאה, "thinking that maybe an invoice was zero or basically an incorrect value". הלקח: על כל כשל גישה או

access denied צריך להחזיר לסוכן הסבר ברור. כך הוא יכול לעצור בצורה מבוקרת, להעביר את העניין לאדם, או לפנות למסד נתונים מוסמך אחר.



"Valid findings, but double counted": ממצאים של כמה סוכנים שרצו במקביל, כל אחד על מנוף חיסכון אחר, בסך כולל של $2.9M שחלקו נספר פעמיים

לקח 4, איחוד ממצאי החיסכון: לצוות יש ספרייה של שיטות לזיהוי חיסכון שנצברה לאורך כ-20 שנה, למשל ספקים משותפים לשתי החברות, או ספקים שונים לאותו שירות. הם הריצו במקביל סוכן לכל שיטה, והתוצאות היו טובות. אבל צריך שלב שמסיר כפילויות, כדי שאותו חיסכון מאותו ספק לא ייספר פעמיים. שלב ההתאמה בנוי על DSPy, והוא מוגדר מראש: על אילו נתונים מסתכלים, איך מבצעים את ההתאמה ומה מותר לעשות אחר כך. Kevin משווה את זה להרצאה אחרת על שימוש ב-DAGs לביצוע מדיניות.

לקח 5, לדעת מה העסק כבר עושה: לדברי Kevin זה הלקח הנפוץ ביותר, ובדיעבד גם המובן מאליו. הסוכן הציע המלצות מצוינות, "but it would be shot down immediately because, hey, we already started that three months ago.". הקושי הוא לתת ל-Claude את ההקשר הנכון בשלב הנכון: לא להגביל את הניתוח מראש, אבל גם לא להגיע להמלצות לא רלוונטיות. הפתרון שהוא מציע הוא לצבור ידע לאורך הפרויקט, בזיכרון או בתוך הכלים, כדי שהסוכן לא יבזבז tokens על גילוי מחדש של מה שכבר ידוע.

7. מה לעשות עכשיו

- **לארגן את הנתונים לפני שהסוכן מתחיל:** להגדיר מראש את השדות שהניתוח יצטרך (תעריפים, הנחות, תאריכי חידוש) ולבנות pipeline שממלא אותם. בלשון המרצים, להשקיע tokens, מאמץ וכסף מראש.
- **לוודא שאף מסמך לא נעלם בשקט:** לספור כמה מסמכים נכנסו לעיבוד וכמה הגיעו למסד הנתונים, ולא להסתפק בתפיסת timeout וממשיכים הלאה.
- **לבנות מחדש את הקשרים בין החוזים:** MSA, SOW ו-amendment, וללמד את המערכת מתי תיקון מחליף את השווי הקודם במקום להוסיף לו.
- **לכתוב חוקים עסקיים בתוך הכלים:** למשל, תאריך הודעה מחושב תמיד לפי סעיף החידוש. לאכוף בקוד ולא רק ב-prompt.
- **להחזיר שגיאות מפורשות:** כשל גישה צריך להחזיר שגיאה ברורה, אף פעם לא אפס או ערך ריק.
- **להוסיף שלב איחוד מוגדר מראש לסוכנים שרצים במקביל:** כדי שאותו חיסכון לא ייספר פעמיים.
- **לתת לסוכן את מצב העסק:** אילו יוזמות כבר בתהליך, דרך זיכרון או בתוך הכלים. כדברי Kevin, "context is still king".

שאלה פתוחה	למה זה חשוב
כמה זמן ועלות חסך ה-pipeline לעומת עבודה ידנית?	ההשוואה שנאמרה היא הערכה בלבד, בלי מדידה
האם מודלים עתידיים ייתרו את החילוץ המובנה מתרשימים?	Kevin מעריך 6–12 חודשים, אבל מסייג ב-"we'll see"

שאלה פתוחה	למה זה חשוב
איך מכניסים הקשר עסקי בלי להגביל את הניתוח מראש?	הוצג כבעיה קשה, בלי פתרון מלא
איך בדיוק נאכפת חסימת הפעולות בשרתי ה-MCP?	תואר ברמת העיקרון בלבד
האם מודלים עם הקשר ארוך מאוד או RLMs מחליפים את הצורך במבנה?	לדבריהם הם לא פותרים את חוסר ההקשר העסקי

8. מילון מונחים

מונח	פירוש
Procurement	רכש: ניהול הספקים, החוזים וההוצאות של הארגון
MSA (Master Service Agreement)	הסכם מסגרת עם ספק, שתחתיו נחתמים הסכמים ספציפיים
SOW (Statement of Work)	הסכם עבודה ספציפי במסגרת MSA
Amendment	תיקון לחוזה. לפעמים מחליף את התנאים הקודמים לגמרי
Orphan amendment	תיקון שלא קושר לחוזה האב שלו
Auto-renewal	חידוש אוטומטי של חוזה אם לא נשלחה הודעה עד מועד מסוים
CLM (Contract Lifecycle Management)	מערכת לניהול חוזים לאורך כל חייהם
OCR	זיהוי טקסט מתוך תמונה או מסמך סרוק
DSPy	framework לבניית תהליכים מובנים עם מודלי שפה, כולל פלט מובנה
Rate card / rate tiers	טבלת תעריפים ומדרגות תעריף
Rebate	הנחה רטרואקטיבית לפי היקף רכישה
MCP	פרוטוקול לחיבור סוכנים לכלים ולמקורות נתונים
DAG	גרף מכוון ללא מעגלים, שמשמש להגדרת שלבי עבודה קבועים
RLM	Recursive Language Models (פענוח משוער של "rlms" בתמליל): גישה לעבודה עם הקשר ארוך מאוד
Warehouse	מחסן הנתונים המרכזי שמולו נבדקו התשובות