

I

Co se to
vlastně stalo?

Po této kapitole umíš vlastními slovy říct, co jazykové modely doopravdy změnily — cena psaní kódu spadla skoro na nulu a hodnota se přestěhovala jinam, do úsudku. A budeš tušit, proč table kniha tvrdí, že bolest, kterou ti stroj ušetří, nebyla daň za porozumění — ona tím porozuměním byla.

Slabý člověk + stroj + lepší proces byl silnější než samotný silný počítač a, což je pozoruhodnější, silnější než silný člověk + stroj + horší proces.

— Garri Kasparov, „*Weak human + machine + better process was superior to a strong computer alone...*“ · *The Chess Master and the Computer*, The New York Review of Books, 11. 2. 2010

Kentauři z New Hampshiru

V květnu 1997 porazil počítač Deep Blue mistra světa Garriho Kasparova a svět uspořádal pohřeb šachu. Titulky hlásaly, že člověk prohrál s myslí ze železa; komentátoři psali nekrology královské hry. Kasparov to viděl střízlivěji: Deep Blue nebyl inteligentní o nic víc než programovatelný budík — jen uměl přepočítat dvě stě milionů pozic za vteřinu a přehrát ho hrubou silou. Přesto se zdálo, že otázka je zodpovězená: v šachu už člověk stroji není soupeř.

O osm let později se ta otázka ukázala jako špatně položená. Roku 2005 se na serveru Playchess konal takzvaný *freestyle* turnaj PAL/CSS: hrát směl kdokoli s čímkoli. Velmistři se svými notebooky, celé týmy analytiků, samotné šachové programy, dokonce specializovaný superpočítač Hydra. Všechny dělící čáry padly — člověk, stroj, tým, na tom nezáleželo. A vyhrál tým, který tam nikdo nečekal: dva amatéři z New Hampshiru, kteří si říkali „ZackS“. Steven Cramton s ratingem 1685 a Zackary Stephen s 1398 — v žebříčku obyčejní hráči, žádní velmistři — a tři úplně běžné počítače.

Jak to dokázali? Ne silnějšími programy; jejich enginy (Fritz, Shredder, Junior) měl kdekdo. Ne lepší hlavou; velmistři je v šachu převyšovali o celé propasti. Vyhráli *procesem*. Naučili se, kdy programu věřit a kdy ne; kdy nechat počítat dál a kdy zasáhnout vlastním úsudkem; jak tři stroje spustit proti sobě a číst jejich neshody jako varování. Ve finále takhle porazili velmistra Vladimira Dobrova s kolegou přes 2600 bodů poměrem 2,5 : 1,5. Slabší hráči s obyčejným železem — a lepším postupem.

Kasparov z toho vyvodil lekcí, která je epigrafem téhle kapitoly a vlastně celé knihy: *slabý člověk se strojem a dobrým procesem porazí i silný počítač, a co je pozoruhodnější, porazí i silného člověka se strojem a špatným procesem*. Ne síla hlavy, ne síla stroje — rozhodl způsob, jakým se ty dvě věci spřáhly. Pro tuhle spřaženou bytost — člověk a stroj svázání procesem — se ujalo jméno *kentaur*. A to je přesně otázka, na kterou tahle kniha odpovídá: jaký proces z tebe a tvého stroje udělá kentaura, který vyhrává, a ne dvojici, která prohrává líp vybavená.



Musím tenhle příběh dovyprávět poctivě, protože má nepohodlné pokračování a kniha o intelektuální poctivosti si nemůže dovolit ho zamlčet. Kentauři nevládli navěky. Kolem roku 2017 se šachové programy zlepšily natolik, že lidský spoluhráč jim už neměl co nabídnout — každý zásah člověka výsledek jen zhoršil. Slavná partie AlphaZero proti Stockfishi z prosince 2017 (osmadvacet výher, dvaasedmdesát remíz, žádná prohra) to zpečetila: v šachu se kentaur přežil. Čistý stroj předběhl spřaženou bytost.

kentaur s procesem — amatéři ZackS (USCF 1685 + 1398), tři běžné počítače



vyhráli

velmistr + stroj — silný člověk, horší proces



superpočítač sám — jen hrubá síla (Hydra)



jak daleko došel v turnaji (síla výsledku)

Zdálo by se, že tím celé podobenství padá — otvírák knihy sestřelený jednou větou. Jenže ta věta tezi neshazuje, otáčí ji a dělá ji silnější. Poučení totiž nikdy nebylo „člověk a stroj si rozdělí práci takhle a navždycky“. Poučení bylo, že *hodnota se stěhuje* — a stěhuje se dál. V šachu

Prameny: G. Kasparov, The Chess Master and the Computer (NYRB, 11. 2. 2010); reportáže ChessBase z PAL/CSS Freestyle 2005 („Dark horse ZackS wins...“). Ratingy 1685 a 1398 jsou USCF (americké), ne FIDE. Ve finále ZackS porazili GM V. Dobrova (+ kolega 2600+) 2,5 : 1,5; enginy Fritz, Shredder, Junior.

kentaur — člověk a stroj spřažení procesem. Metafora knihy: hodnota není ani v člověku, ani ve stroji, ale v tom, jak jsou svázání. Křtí se tedy; vrací se v celé druhé části.

Freestyle 2005: nevyhrál nejsilnější člověk (velmistr + stroj) ani nejsilnější stroj (superpočítač Hydra sám), ale kentaur s lepším procesem — dva amatéři s obyčejnými počítači. Sanguine sloupec je vítěz. Titul ani hardware nerozhodly; rozhodl postup.

se přestěhovala z hrubého počítání (Deep Blue) do procesu spřažení (kentaury) a odtud zase do stroje samého. Konkrétní dělba práce mezi člověkem a strojem je dočasná; co přežívá napříč všemi těmi přesuny, je jediné: kdo má lepší proces a lepší úsudek, kam ho vložit, vyhrává tu partii, která se zrovna hraje. A my zrovna teď stojíme na začátku úplně nové partie — ne v šachu, ale ve všem, co se dá napsat do textu. Tam se hodnota právě teď stěhuje znovu, a tahle kniha je o tom, kam.

Jedna noc, kterou pamatuju dodnes

Než dojdeme k tomu, kam se hodnota přestěhovala, musím ti něco přiznat — protože zrcadlo téhle knihy ukazuje nejdřív autora, ne čtenáře. Víš, o čem mluvím, protože jsem to viděl zblízka a mrzí mě to dodnes.

Znám kluka — syna kamaráda — který jednou seděl celou noc nad Excelem. Měl v jednom souboru deset tisíc řádků objednávek a úkol, který zněl nevinně: zjistit, kolikrát se která položka opakuje. Nevěděl, jak na to jinak než ručně — najet myší, najít, přepsat, přičíst, další řádek. A tak seděl. Když jsem ho ráno potkal, byl bílý jako stěna, měl hotovou možná pětinu a prošvihl odevzdání. Ráno by mu byl stačil jeden příkaz a pár vteřin. Té noci jsem u toho nebyl, abych mu to ukázal — a proto ji tady vyprávím.

Nejde o tu jednu noc. Ta je dávno pryč a kluk se z ní vyspal. Jde o *všechny další* takové noci — o celé roky práce, kterou tenhle svět odpracoval myší, protože nikdo neviděl, že úloha má jiný tvar. A jde o rozhodnutí, které ta noc udělala za kluka, aniž to tušil: postavila ho na jednu stranu čáry, která tímhle světem prochází.

Svět se totiž dělí na dva druhy lidí. Na ty, kdo se na výsledek podívají a *poznají*, že je špatně — že to číslo nemůže sedět, že tam chybí řádek, že se to nemá jak stát. A na ty, kdo se musejí spolehnout, že to někdo nebo něco udělalo správně, protože sami nemají jak to ověřit. Kluk nad Excelem byl v té druhé skupině: kdyby se v tom nočním přepisování o dvě stě řádků spletl, nikdy by se to nedozvěděl — neměl proti čemu výsledek porovnat. A teď to důležité, protože tahle kniha není o tom, koho do které skupiny zařadit: *do té první skupiny se dá přejít*. Ta čára není hradba, jsou to dveře, a tahle kniha je vstupenka skrz ně. Motto, které nese celou knihu, zní: každý může být vědec a programátor. Ne každý jím je — ale každý se jím může stát, a tady je návod.

A teď to nejdůležitější, jádro celé teze knihy, a čti to pomalu. Ta bolest — noc nad Excelem, roky odpracované rukama, dřina, o které si dnes myslíme, že byla zbytečná — *nebyla daň za porozumění*. Ona *tím porozuměním byla*. Kdo protrpěl dost takových nocí, přestal je jednou provždy trpět: naučil se v úloze poznat její tvar, naučil se čekat, kde se výsledek pokází, naučil se, že tohle číslo nemůže být správně. Ta

„Svět se dělí na dva druhy lidí“ v téhle knize nikdy nekončí posměškem. Za každou takovou větou jsou dveře: ne kdo je uvnitř a kdo venku, ale kudy se dovnitř jde. Vstupenka, ne hradba. Zapamatuj si to — potkáš tu figuru ještě mnohokrát.

schopnost — poznat, že je něco špatně — nespadla z nebe. Vyrostla z jizev. Bolest byla škola, ne poplatek za vstup do ní.

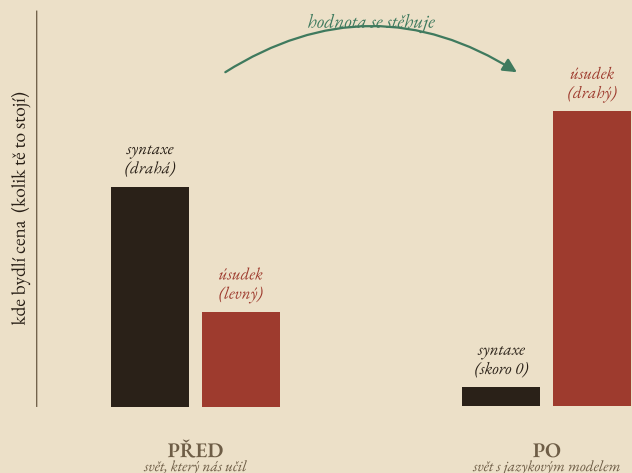
A přesně tuhle bolest teď stroj bere pryč. To je ta dobrá zpráva i ta zrada v jedné větě.

A at' je od začátku jasné, na čí straně ta kniha stojí: nástroje jsou *skutečně úžasné*. To, co dnes napíše jazykový model za vteřinu, mě kdysi stálo noci. Nepíšu chvalozpěv na staré dobré časy a nechci ti brát stroj z ruky — píseň těch sirén je opravdu krásná a máš právo ji poslouchat. Jde jen o to nepřijít při tom o to jediné, co ti ta krása tiše odnáší: o úsudek, který ses jinak učil právě tou dřinou, co teď mizí.

Kam se ta hodnota přestěhovala

Řekněme to teď přesně, bez metafor, protože na téhle jedné větě stojí celá kniha. *Jazykové modely srazily cenu syntaxe skoro na nulu — a hodnota se přestěhovala do úsudku.*

Rozeberme to na dvě části. „Syntaxe“ je ta viditelná, pracná půlka programování, o které si laik myslí, že *je* programování: pamatovat si, jak přesně se píše ten příkaz, kde je středník, jak se jmenuje ta funkce, jaké přepínače má ta roura z Excel noci. Tuhle půlku jsme se dřív učili roky a platili za ni nocemi — a přesně tuhle půlku dnes stroj udělá za tebe, správně a okamžitě. Cena syntaxe padla skoro na nulu. To je ta krásná zpráva a není na ní nic falešného.



Vlevo svět, který nás učil: většina ceny byla v syntaxi (napsat to správně), úsudek přišel skoro zadarmo k tomu. Vpravo dnešek: syntaxi napíše stroj skoro zdarma, a celá cena se přesunula do úsudku — poznat, co chtít a jestli to, co přišlo, je správně. Celková hodnota se nezměnila. Jen se přestěhovala.

Ta druhá půlka je *úsudek*: vědět, co vlastně chceš, aby program dělal; poznat, že výsledek, který ti přišel, je špatně; tušit, kde se to celé zhroutí, až přijde jedenáctitisící řádek. Tahle půlka nikdy nebyla vidět, protože se schovávala uvnitř té dřiny — kdo protřpěl syntaxi, dostal úsudek jaksi mimochodem k tomu. A právě tady je ta past: když teď odpadne dřina,

odpadne i ta bezplatná škola úsudku, kterou v sobě skrývala. Stroj ti dá syntaxi zdarma, ale úsudek ti nedá — ten si musíš pořídit jinak, protože cesta, kterou k němu lidé chodili dvacet let, se právě zavřela. Podívej se na ten obrázek ještě jednou: hodnota z levého světa nezmizela. Přestěhovala se doprava, do sloupce, který je teď najednou vysoký — do úsudku.

A tady se otvírá ta nůžková mezera mezi dvěma skupinami lidí, o kterých jsem mluvil u Excelu. Kdo úsudek má, dostal do ruky nástroj, který mu ho tisíckrát zesílí: řekne si o rouru, dostane ji, a protože *pozná*, jestli je správná, letí. Kdo úsudek nemá, dostal tentýž nástroj — jenže ten mu tisíckrát rychleji vyrobí výsledek, o kterém neumí říct, jestli je dobrý, a ještě mu k němu pográtuluje. Stejný stroj, opačný osud. Rozdíl není v ceně nástroje — ta je pro oba stejná, skoro nulová. Rozdíl je v úsudku, a ten je teď to jediné vzácné.

Co se děje mezi promptem a kódem

Pojďme to osahat konkrétně, protože „úsudek“ zní vznešeně a mlhavě, dokud neukážeme, kde přesně v práci sedí a kde ho žádný stroj neudělá za tebe.

TEENAGER · MECHANISMUS

Rozeber si takzvanou *vibe coding session* — sezení, kde píšeš stroji, co chceš, a on ti to vyrábí — na tři kroky. Úsudek nesedí ve všech třech stejně:

1. SPECIFIKUJ řekni PŘESNĚ, co má vzniknout
(„spočítej, kolikrát se každá položka opakuje“)
→ úsudek: vědět, co vlastně chci. NELZE delegovat.
2. NECH VYGENEROVAT stroj napíše kód / rouru / text
→ syntaxe: tohle stroj umí líp než ty. DELEGUJ.
3. OVĚŘ NEZÁVISLE poznej sám, jestli je výstup správně
(ne „vypadá to dobře“ — otestuj to jinou cestou)
→ úsudek: poznat, že je to špatně. NELZE delegovat.

Krok 2 klidně přenech stroji — je v něm lepší. Ale kroky 1 a 3 jsou tvoje a nikdo ti je nesundá z krku, protože to jsou přesně ta dvě místa, kde bydlí úsudek. Kdo přeskočí krok 1 („však on nějak pochopí, co chci“) nebo krok 3 („funguje to, tak dobrý“), neděleval práci — vzdal se jí.

Všimni si, že tohle je *kentauroví checklist* z podobenství, přeložený do práce: specifikuj → nech vygenerovat → ověř nezávisle. Amatéri na Playchess vyhráli právě proto, že měli krok 3 zvládnutý — věděli, kdy stroji nevěřit a jak jeho výstup nezávisle prověřit. Kdo z těch tří kroků udělá jen ten prostřední, není kentaur; je jenom rychlejší dvojína, který navíc netuší, kam střílí.

A teď ta nejnebezpečnější věta celého řemesla, věta, kterou uslyšíš od sebe i od druhých pořád dokola: „**Funguje to.**“ Zní jako závěr, ale je to jen dojem — a dojem není důkaz. „Funguje to“ znamená obvykle tohle: *zkusil jsem jednu věc, dopadla, jak jsem doufal, a přestal jsem se dívat.* Program, který spočítá výskyty správně na třech řádcích, „funguje“ — a rozsype se na deseti tisících. Odpověď modelu, která zní sebejistě a čte se hladce, „funguje“ — a je celá vymyšlená. Krok 3 existuje přesně proto,

aby porazil větu „funguje to“: neptáš se, jestli to *vypadá*, že to jede, ptáš se, jak bys poznal, že to *nejede* — a hledáš odpověď jinou cestou, než kterou to vyrobil stroj.

Půlvěz: ekonomie jednoho zlevnění

Zatím jsme mluvili obrazem. Teď přitvrdíme — ne až na těžkou matematiku, na tu tady není důvod, ale na poctivou ekonomii, protože ta říká přesně a střízlivě, proč zlevnění syntaxe *nezmenší* objem práce a proč cena úsudku poroste. Vzorce necháme na okraji; v hlavním textu stačí tři pojmy.

První je *Jevonsův paradox*. Když v 19. století zlevnil uhelný pohon (Wattův parní stroj spálil na tutéž práci méně uhlí), ekonom William Stanley Jevons čekal — jako každý — že se spotřeba uhlí sníží. Stal se opak: spotřeba *vzrostla*. Levnější pohon otevřel tolik nových použití, že se ho nakonec spálilo mnohem víc. Zlevnění nesnížilo poptávku, rozšířilo ji. Přenes to na dnešek: když psaní kódu zlevní skoro na nulu, nebude se psát *méně* programů — bude se jich psát mnohonásobně víc, protože se najednou vyplatí naprogramovat i věci, na které si dřív nikdo netroufl. A každý ten nový program potřebuje někoho, kdo pozná, že je špatně. Práce úsudku nezmizí; naliže se jí víc.

Druhý pár pojmů je *substitut* a *komplement*. Substitut je věc, kterou nová věc *nahradí* — máslo za margarín. Komplement je věc, které nová věc *zvedne* cenu, protože se bez ní neobejde — auto zvedlo cenu benzínu. Jazykový model je substitut *syntaxe*: nahradil pracné psaní příkazů, a cena syntaxe proto padá k nule. Ale je *komplementem úsudku*: čím víc kódu stroj vychrlí, tím cennější je ten, kdo pozná, který z něj je k ničemu. A cena komplementu s rozšířením té laciné věci *roste*. Odtud plyne celá strategie téhle knihy jednou větou: nekupuj si to, co zlevnilo (syntaxi umí každý stroj), ale to, co kvůli tomu podražilo (údek).

Třetí věc není pojem, ale výstraha — a týká se toho, jak číst studie, které ti budou tvrdit, o kolik jsi teď rychlejší. Roku 2025 provedla skupina METR kontrolovaný pokus: šestnáct zkušených vývojářů, čtyřicet desítek úkolů na projektech, které roky znají. Půlku úkolů směli dělat s pomocí AI, půlku bez. Výsledek se vzpírá intuici: s AI byli vývojáři o devatenáct procent *pomalejší*. A teď to podstatné, kvůli čemu tu ten pokus je: *oni sami si mysleli, že jsou o dvacet procent rychlejší*. Pocit rychlosti ukázal opačným směrem než měřený čas. To je celá kniha v jednom datovém bodě: dojem („funguje to, letím“) a skutečnost („měřeno, brzdím“) se rozešly — a bez měření bys věřil dojmu.

Nečti to jako „AI je k ničemu“ — to by byla stejná chyba jako pohrbitvat šachy po Deep Blue. Čti to takhle: pocit dřiny ani pocit rychlosti nic nedokazují, a jediná měna, která platí, je měřitelný výsledek. Kentauři z New Hampshiru nevyhráli proto, že se *cítili* silní; vyhráli,

→ kap. 8: kroky 1 a 3 zapsané předem (*co chci, a jak poznám, že je to špatně*) jsou první pakt — *smlouva se sebou samým dřív, než otevřeš chat*.

→ kap. 22: „ověř nezávisle“ *dorůstá do celé dovednosti* — kalibrovaná důvěra: *věřit úměrně tomu, jak levné je ověření*.

☞ EINSTEIN

Jevonsův paradox (Jevons, The Coal Question, 1865): *klesne-li cena vstupu, poptávané množství vzroste tak, že celková spotřeba stoupne, ne klesne. Zlevní-li psaní kódu na zlomek, poroste množství napsaného kódu rychleji, než cena klesla — a s ním i potřeba ho ověřovat.*

☞ EINSTEIN

Substitut vs. komplement. Zlevní-li A, klesne poptávka po jeho substitutu a stoupne po jeho komplementu. Model je substitut syntaxe (→ cena syntaxe ↓) a komplement úsudku (→ cena úsudku ↑). Tvá páka je v komplementu.

☞ EINSTEIN

METR 2025 (arXiv:2507.09089): 16 zkušených vývojářů, 246 úkolů, RCT. S AI o ≈ 19 % pomalejší; sami odhadovali, že jsou o ≈ 20 % rychlejší. Malý vzorek, zkušení lidé na známých projektech — ne verdikt „AI zpomaluje všechny“, ale tvrdý doklad, že pocit rychlosti není důkaz rychlosti.

protože měli proces, který jim řekl, kdy se jejich pocit mylí. Přesně tenhle proces se teď učíš i ty.

Co si z první kapitoly odnést

Poskládej to dohromady, protože tohle je odrazový můstek do celé knihy. Kentauři ukázali, že o vítězství nerozhoduje síla člověka ani síla stroje, ale proces, který je spřáhne — a že hodnota se stěhuje dál, jakmile se dělba práce změní. Excel noc ukázala, že bolest, kterou dnes stroj bere pryč, byla tou školou, ze které vyrostl úsudek. Ekonomie ukázala, proč zlevnění syntaxe práci nezruší, jen ji přesune — a proč se cena stěhuje z toho, co stroj umí, do toho, co poznat neumí. A METR ukázal, že vlastnímu pocitu rychlosti se nedá věřit bez měření.

Zůstává jedna otázka, na kterou tahle kniha odpovídá dvaadvaceti kapitolami: když už tu dřinu nemusíš protrpět, *kde vezmeš úsudek, který ti dřív dala?* Krátká odpověď je metoda — vědomá náhrada bolesti, stěžeň, ke kterému se přivážeš dřív, než tě píseň sirén svede přes palubu. Tu metodu budeme stavět kámen po kameni; její stěžeň vztyčíme v kapitole osmé a její jádro pokřtíme v kapitole deváté. A rozuzlení Excel noci — ten jeden příkaz, který měl kluk napsat místo celé noci — na tebe čeká v kapitole sedmnácté, kde uvidíš, že nešlo o to neznat syntaxi, ale nevědět, že to jde.

Ale jedno vlákno té metody potřebuješ hned, od první stránky, protože bez něj je zbytek knihy jen sbírka příběhů. Je to jediná otázka, kterou se liší člověk, co drží lesk, od člověka, co drží bídu — a od téhle chvíle si ji budeš klást u každého výsledku, který ti stroj vyrobí.

→ kap. 8: metoda jako stěžeň — pakt se sebou samým, uzavřený za strážliva.

→ kap. 17: rozuzlení Excel noci — najdi | seřaď | spočítej; jeden řádek místo celé noci.

→ kap. 9: refrén této knihy se tam křtí — tady ho zasazujeme.

„Jak bych poznal, že se pletu?“

Tenhle blok bude od teď končit každou kapitolu; naplno se pokřtí v kapitole deváté, ale zasazuje se tady, protože ho budeš potřebovat dřív. Konkrétní test k první kapitole: vezmi úplně příští věc, kterou ti AI vyrobí — kód, tabulku, odstavec, cokoli — a než ji použiješ, odpověz si na jedinou otázku: *jak bych poznal, že je špatné?* Ne „vypadá to dobře“ — to je dojem. Chci konkrétní, ověřitelnou odpověď: který řádek bych zkontroloval, které číslo spočítal jinou cestou, který okrajový případ vyzkoušel. Tvrdím, že když na tu otázku nedokážeš odpovědět něčím konkrétním, nevíš, jestli držíš lesk, nebo bídu — máš výsledek, kterému musíš jen věřit, a to je přesně ta druhá strana čáry. A když odpovědět dokážeš a tu kontrolu uděláš, právě jsi udělal krok tam, kam celá kniha míří: z těch, kdo věří, mezi ty, kdo poznají.