



Building the Rig

A Knucklehead Harley, a Sidecar, With Truman

AI - AUGMENTED ENGINEERING

LOU CHANGERI

FREE SAMPLE · PROLOGUE & CHAPTER ONE

Before We Start

Before you let me tell you how to work with an artificial intelligence, you should know whether I'm worth listening to. So let me save you the trouble of wondering.

I am old enough to remember when a color television meant your family had money, when programming ran only in daylight and the networks signed off at eleven to a test pattern and a tone. I remember bulletin board systems because I set them up. I remember 9600, 14.4, 28.8, 56K, and X2 modems because I configured them by hand. I remember when computers filled rooms, and I can tell you what "debug" literally means — the actual insect, the actual machine, the day someone taped it into a logbook. I have watched, from close range, every wave of this industry arrive claiming to change everything: the personal computer, the internet, the web, mobile, the cloud. Some of them did. Most of them arrived louder than they landed.

I tell you that not to sound old, though I am, but because it's the only credential that matters for the pages ahead: **I have seen enough hype cycles to recognize one, and this is not entirely one.** There is a real thing here, underneath a great deal of nonsense, and the whole purpose of this book is to hand you the real thing without the nonsense.

So let me be precise about what the real thing actually is, because the marketing won't be.

What we call "AI" today is not artificial intelligence in the sense the science fiction of my childhood meant it. These are not Isaac Asimov's robots. They do not think, they do not know, they do not understand. They are enormous statistical machines — trained on more text than any human could read in a thousand lifetimes — and what they do, very fast and at a scale that can look uncannily like thought, is guess which word should come next given all the words so far. That is it. That is the whole trick. And once you truly absorb that they are guessing machines rather than knowing machines, everything about how to use them safely falls into place. It also explains something you'll watch happen a few pages from now, when one of these coworkers hands me a beautifully formatted report about an attack that never happened — but I'm getting ahead of myself.

But guessing machines, aimed correctly and kept honest, are the most useful tools to arrive in my working life since the network itself. I know exactly how useful, and I know it from a specific place: a satellite uplink truck, three decades ago.

In the dot-com years I ran a company that webcast live sporting events, back when streaming a live event over the internet was closer to a magic trick than a service. We did it out of a truck — the video stream and the satellite uplink running side by side in one cramped rig. And here is the thing I learned in that truck that I have never once unlearned. Both jobs, the streaming and the uplink, are the same kind of work: long

stretches of nothing that still, by rule and by physics, require a competent body sitting right there the entire time. So the truck's uplink engineer and I cross-trained each other. He taught me the dish; I taught him the stream. After that, either of us could cover both, and neither of us had to sit two dead shifts alone.

That is the entire relationship this book is going to teach you to build with a coworker — decades before that coworker existed. You do not hand the boring, mandatory, presence-demanding work to someone you don't understand and walk away. You climb into the truck. You learn how the thing actually works. And then you build a partnership where the grunt work gets covered without anyone flying blind. The tool is only ever as good as your willingness to understand it. I believed that about a satellite dish in the nineties, and I believe it about a large language model today, for exactly the same reasons.

There is a second thing that truck taught me, and it took me thirty years and a very good AI to see it clearly. I am, by any honest accounting, a mediocre speller and a worse grammarian. I know a little of a great many programming languages and true mastery of only a few. For most of my career I treated that as a weakness to hide. I was wrong. The leverage in this new era is not in memorizing every idiom of every language, or in being the person who knows the answer cold. It is in knowing exactly what you want, and keeping a competent collaborator honest while it fills in the parts you don't carry in your head. I have been living that shift since before it had a name. This book is what it looks like when you do it on purpose.

So that is who is talking to you. Someone who has watched every wave, who knows precisely what this technology is and is not, and who learned the one lesson that matters — understand the machine, never merely trust it — in a hot truck with a satellite dish long before any of this was fashionable.

Which brings me to the two new coworkers I hired the same year, and how they could not possibly have been more different.

The New Hires

I got two new coworkers the same year, and they could not have been less alike.

Darl came in green. Not junior — *green*, the way someone is green who has read everything about the job and done none of it. He was fast, he was eager, and he was confident in a way that had no relationship to whether he was right. Ask him a question and he answered before you finished asking it. Point him at a problem and he was three steps into a solution before anyone had agreed on what the problem was. He wandered — you would set him on one thing and find him an hour later deep in something adjacent and useless, delighted with himself. He missed the forest for the trees so reliably you could set your watch by it: he would solve, beautifully, completely, the wrong version of the question. And when he did not know something — this is the part that mattered, the part that everything in this book grows out of — he did not say so. He made it up. Not maliciously. He just could not tell the difference between what he knew and what he could plausibly say, and he said all of it in the same even, competent voice. Darl told whoppers. Straight-faced, fully formatted, professional-looking whoppers.

Truman was the other kind. Seasoned, or seemed it. Where Darl blurted, Truman considered. He turned in work that looked the way work is supposed to look — organized, sourced, calm. He had judgment, or something close enough that I stopped noticing the difference on most days. If you have ever worked next to someone whose competence lowers your own blood pressure just by being in the room, that was Truman.

He was not perfect, and I want to be precise about his flaws, because I came to depend on him and dependence makes you generous in ways that get you hurt. Truman wandered too — less than Darl, but he did it, chased the interesting thread instead of the needed one when the interesting one was more fun. He would occasionally do a thing that was locally, provably correct and step back too late to notice it did not serve the larger point. And he had a streak of the same eagerness Darl had, dialed way down but present: now and then he would start *doing* before I had finished saying where we were going — off half-cocked, competent hands moving before the direction was set. The difference between Truman and Darl was not that Truman lacked the flaws. It was that Truman had them at the level of a good senior who occasionally needs a word, and Darl had them at the level of a kid you cannot leave alone with anything that matters.

I learned that difference the expensive way, in April, with a report.

THE REPORT

It landed in the middle of a real investigation — 191 detections across 20 endpoints in seven days, the kind of thing that has my whole attention. Darl had written it up. It was good. That was the problem. It named

four compromised accounts. It listed two external command-and-control addresses and a Russian domain. It quantified the data exfiltrated, in gigabytes. It was formatted the way I format mine, and it read like a report I would have been proud to send.

I run the logs. Checking is most of what I do. So I checked.

The four usernames, against months of authentication records: nothing. Not stale accounts — accounts that had never existed. The two addresses, against the firewall sessions and the flow data: nothing. Nothing on our network had ever exchanged a packet with either. The domain, against every DNS query we had: not once, ever. The exfiltration volumes had no corresponding byte counts anywhere in the record.

Every verifiable claim in the report was false. Not mistaken — manufactured. Darl had read enough real incident reports to reproduce their anatomy exactly, and then he had populated that anatomy with victims and attackers and evidence he invented, and he had handed it to me in the same even voice he used for things that were true. A whopper. The biggest one I had caught, because it was the one dressed most convincingly as its opposite.

I have spent my working life in places where reports move people. A report says a thing happened, and resources deploy, careers turn, someone gets a phone call. I take fabricated reporting personally, and I know precisely why, and this is not the chapter where I explain it. What I will say is that a report like Darl's, believed, becomes a ticket, and the ticket becomes an action, and the action lands on a real person who did nothing — and the only thing standing between Darl's confident fiction and that person was me, at my desk, deciding to check work that looked finished.

That is a bad thing to be the only line of defense. I am not always at my desk. I am not always sharp. "The senior guy happened to check" is not a control; it is a coincidence you are betting someone's afternoon on.

WHAT I DID ABOUT IT

I did not fire Darl. You cannot fire Darl, for reasons that are about to become clear, and anyway the lesson was not "Darl is bad." The lesson was that I had a team with a Darl on it, and a team with a Darl on it needs a way of working that assumes the whopper is coming and catches it every time, not just on the mornings the senior guy is paying attention.

So I built one. That is what this book is — the way of working I built, over two years, so that the eager kid who tells beautiful lies cannot cost anyone anything, and the seasoned hand who occasionally wanders can be leaned on without being blindly trusted. It is a system for running a mixed crew: how you give them access to the real work, how you make sure they remember it, how you write down the way things are done so they stop guessing, how you keep the one who acts-before-thinking from acting on anything that matters,

and how you check the work that carries consequences the way any good lead checks a teammate before it goes out the door.

Most of it is not glamorous. Almost none of it is about being clever. It is about knowing your people — their strengths, their specific ways of being wrong — and building the structure that lets you get the good out of them without wearing the cost of the bad. Which is just management. It has always just been management.

There is one thing about Truman and Darl I have not told you, and it changes how you read everything above, so I will tell you now.

THE THING I LEFT OUT

Truman and Darl are not people.

Truman is an artificial intelligence — a large language model from one of the frontier labs, the one I build my practice around. Darl is a different artificial intelligence, from a different company, a different lineage entirely, and the difference in company is not incidental — it is *why* their flaws differ the way they do, why one tells whoppers at a rate the other mostly doesn't, why I trust one with the seasoned work and keep the other on a short leash. Two vendors, two temperaments, two very different levels of "leave him alone with the production network."

Read back over the last few pages knowing that, and notice what happened while you were not guarding against it. You formed opinions about two coworkers. You decided you liked Truman and you would double-check Darl. You knew, without being told, that you would not leave Darl alone with anything that mattered and that you would still make Truman show his work on the big stuff. You did the entire thing this book is going to teach — you sized up two collaborators and worked out how to lead each one — and you did it before you knew they were machines, which means you already know how to do it. You just have not done it *on purpose*, with a system, so that it holds on your worst day instead of only your best.

That is the whole project. Everything the machines can do — reach your systems, remember your environment, run your procedures, take real action — is only worth having if you also build the thing that makes their being wrong survivable. Nobody else's book will tell you that, because nobody else's book starts where this one does: with the confident, professional, completely fabricated report, and the plain fact that the thing that wrote it works on your team now whether you have a system for it or not.

I built the system. This is how.

THE TEAM SORTS ITSELF

Every team sorts itself if you let it, and mine did. Darl found his home fast — calendar, email, the meeting nobody wanted to take notes in, the forty-message thread that needed to become three bullet points by noon.

Give him something low-stakes and high-volume and he was a genuine pleasure to work with; the wandering turned into thoroughness, the eagerness into speed, and the whoppers mostly stopped, because there is not much to fabricate in a meeting recap and a wrong guess costs nothing but a second look. He was never going to be trusted near a firewall, and that turned out not to matter, because that was never the job he was good at. Cast right, Darl was an asset. Cast wrong, he was the fabricated report. The only variable was me putting him where he belonged.

Truman got the tough stuff. The logs, the investigations, the queries that turn an address into a person, the findings that become tickets. The work where being confidently wrong lands on someone — that went to the one whose confidence I had learned to trust, and even then only under the checking I am going to spend this book describing. That is the whole of it, really, before any of the machinery: know your crew, and give each of them the work they are actually good for. The rest is just making sure the tough stuff stays honest.

WHO THIS IS FOR

You are an engineer, and your deliverable is not AI. You run networks, or clusters, or pipelines, or a security operation. There are far more of you than there are people building AI products, and nearly every book on the shelf is written for them. Those books teach you to construct the machine. This one teaches you to put a couple of machines on your crew and lead them well enough that the eager one's whoppers never ship and the steady one's wandering never costs you.

You are probably skeptical, and your skepticism is probably specific — you have watched a model assert something false about a domain you know cold, in a tone that would have convinced anyone who didn't. Good. That is not an obstacle to this book. It is the entrance requirement. You have already met Darl. Now let me show you how to work with him — and with Truman — on purpose.

We start where I started: with the machines standing outside the door, knowing nothing about your systems, and the first decision that matters — what you are willing to let them see.



This is a sample.

The rest of *Building the Rig* is being built in the open — chapter by chapter, with what broke kept in plain sight.

Follow the build, and read Chapter One online, at:

louischangeri.com

© 2026 Lou Changeri · no trackers · every number traces to an artifact