

Structural CTO

Naming the work you already do

Anthony S. Jackson, MBA

Chapter 1, shared with The Load Curve™ guide

Chapter 1: The Load Curve

In structural engineering, a load curve plots an applied load against a structure's response across a range. Engineers use it to find something specific: the point where a structure's behavior changes character, not just scale. A beam under light load bends predictably. Add weight steadily, and at some point, before failure, the beam starts absorbing load differently than it did ten pounds earlier. The curve shows you where that shift happens. Without it, you're guessing at a straight line where the real shape has bends in it.

Every fractional CTO firm I researched sells by company size. Seed stage. Series A. Ten million in revenue. Fifty million in revenue. It's an easy axis to market against, and it's the wrong one, because it assumes a company's need for CTO-level structural thinking scales in a straight line with headcount or revenue. It doesn't. It climbs, flattens, and climbs again, at points that have almost nothing to do with the number on the cap table.

Picture two companies at the same revenue: eight million dollars a year. One is a twelve-person team building a single product, with a technical founder who still reviews every architecture decision personally. The other is a forty-person team spread across three product lines, two of them acquired, running on infrastructure nobody currently on staff originally built. Same revenue. Opposite position on the curve. The first company doesn't need a Structural CTO yet. The second needed one eight months ago and doesn't know it, because nobody ever told them revenue was the wrong thing to check.

I tried building a five-tier service menu around this once. It read exactly like every fractional CTO staffing page already crowding this market, priced from the bottom up, comparable feature for feature with six competitors before I'd finished the outline. I scrapped it. The curve earns its keep as a diagnostic, pointing you toward what to examine in a prospect's organization, long before you ever price anything.

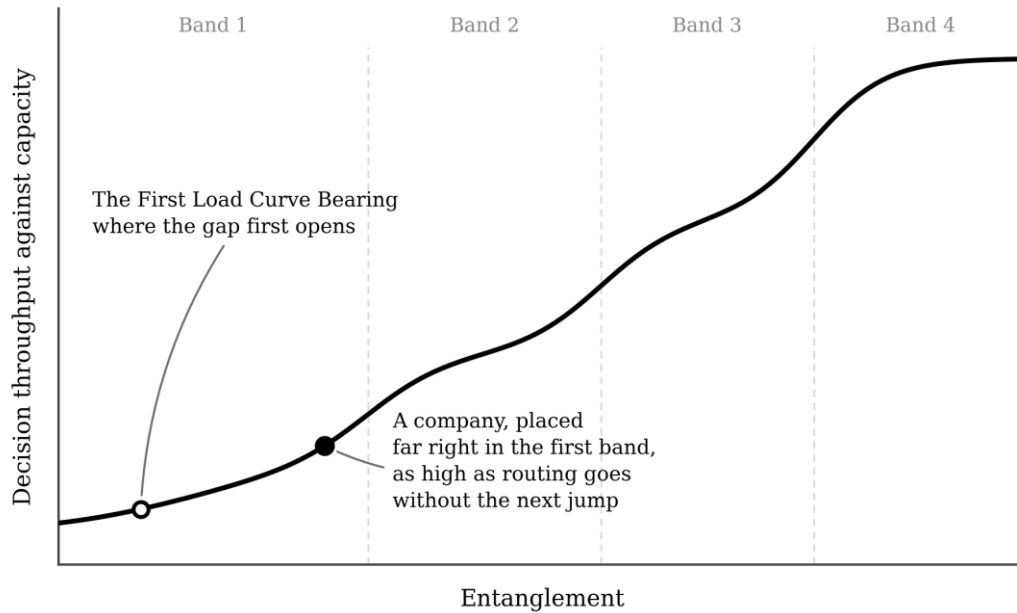
What the Load Curve measures is the gap between two things that rarely move together: how much structural technical decision-making a company's current situation actually requires, and how much of that decision-making the people

already in the room can absorb alongside their existing jobs. Early on, that gap sits close to zero. A capable technical founder or a strong senior engineer can hold the whole architecture in their head, because there isn't much architecture yet. Entanglement opens that gap: more product lines, more legacy decisions, more people making technical calls with no shared model of why the last five calls were made the way they were. Company size is a proxy for entanglement, and a weak one.

A curve has two axes. This one has been described across a whole market without either of them being named, which is why it stayed a metaphor instead of becoming an instrument.

The vertical axis is that gap made countable: decision throughput against capacity. Count the structural decisions in a month that must route through one person before they can move, and count how long each waited. Not every decision. The ones that cross a boundary: architecture, sequencing, build against buy, which commitment the roadmap can actually carry. Both numbers are available to anyone willing to read a calendar and a ticket queue for four weeks.

The horizontal axis is entanglement, and its components are the three already named: how many product lines share a fate, how much of the running system was inherited from people who are no longer there, and how many decision-makers act independently with no shared model of what they are building.



product lines sharing a fate · inherited systems · independent decision-makers

The Load Curve. Entanglement runs along the bottom, decision throughput against capacity up the side. The bands are the stable stretches; the climbs between them are the jumps. Both marked points sit in the first band.

The gap that vertical axis measures is a Structural Bearing™: a permanent tension between two legitimate systems, neither of which can be removed without breaking something the business needs. What the situation requires is legitimate. What the people in the room can absorb is legitimate. The market reads the space between them as a strength problem, answered with a stronger person or more hours from the person already there. The variable is distance, and distance is managed rather than closed.

Now the part that looks like a contradiction and is not.

These jumps tend to happen around ten million, thirty, fifty, and eighty. I am not adopting the number. On average, that is the revenue of a company with certain operational characteristics, and the characteristics are the thing being read. The industry treats the correlate as the cause, which is how you end up with a service tier for Series A companies. Two companies at the same revenue sit in different places, and the placement explains why one is fine and the other is about to stop being fine. The bands are rough on purpose, and different

industries and different players cross them at different points. What does not move is the nature of what changes.

The practices that carry a company into a band are the specific thing that prevents it from leaving. A company does not travel from ten million to thirty on startup practices built around personalities. It has to replace them with systems first. Not because the old practices stop working, but because they are what makes the next stage unreachable.

Which means the structure is not the response to growth. It is the gate on it. Read that direction carefully, because everything about how this work gets sold depends on it. You are not repairing something that broke under load. You are building the thing without which the load never arrives.

Here is one, placed.

A twelve-million-dollar business. Three product lines sharing a fate: a desktop product carrying most of the revenue, a cloud version in early adopter state, and an API sold as a cheap add-on to the desktop. The running system had been built by two managers who were gone before I arrived, and what they knew about why it worked that way left with them. Every decision crossing a boundary between those three products routed through one seat.

Far right on entanglement for a company in the first band, as high on routing as that band goes, and nowhere near the second. The work was replacing personality-centered practices with systems, which is that band's whole nature. Had the API not been promoted to a real product in the middle of it, the next jump would not be available at all. Not delayed. Not harder. Unavailable.

Revenue more than doubled in under two years.

I want to be precise about what that number is evidence of. It is not proof that better structure produces growth, which would be an easier claim and a weaker one. It is what a gate opening looks like. The work was preparation for a jump the company had not made yet, and the one after it will come faster because that preparation is already behind us.

Placing a real company takes one conversation and two sets of questions.

Entanglement first, because a founder answers it from memory. How many product lines share a fate. How much of the running system was built by people no longer on staff. How many people make boundary-crossing calls without checking with each other. Three answers, and the horizontal position is set.

Then the vertical. Ask for the last three decisions that crossed a boundary, and for each one, the date it was raised and the date it was settled. Founders know those dates. The queue time between them is the number, and it is the one nobody has ever added up.

Write the reading down, date it, sign it, and hand it over. Name both halves: the signals present, and the ones you looked for and did not find. The absences separate a reading from a pitch. A signed reading travels, into the board deck and to the co-founder who was never in the room. An unsigned one becomes the founder's own thinking, unattributable by the time it matters.

Here is what one actually looks like. Take a twenty-eight-person software company, six years old, eleven million in revenue, with one technical founder who has never stopped personally reviewing every architecture decision.

Entanglement: one product line, so nothing is competing for the founder's attention on that axis. The founder built the system and still runs it, so nothing has been inherited from anyone who left. One person makes every call that crosses a boundary, and nobody else is authorized to.

Decision throughput: the last three architecture decisions that crossed a boundary took eleven days, fourteen days, and twenty days from the day they were raised to the day they were settled, each one waiting on the same calendar.

Signals present: single-seat authority on every cross-cutting call, queue time climbing across the three most recent decisions, “we should really document this” surfacing unprompted from two different people in separate conversations.

Signals looked for and not found: no second person authorized to make a boundary-crossing call when the founder is out, no written record of why the current architecture looks the way it does, no standing forum where the founder's judgment gets tested against anyone else's.

Dated, signed, two paragraphs long. That is the whole artifact.

When a founder disagrees, find out which half is in dispute. The count settles itself, because the calendar and the ticket queue are sitting right there. What the count means will not settle in that room, and pressing it there costs you the room. The dated reading is what makes that survivable. It stands while the company keeps running, and the next boundary decision that stalls tests it without you present.

You have the curve, both axes, and a way to place a company on it. One point on that curve matters more than the rest: the place where the gap first opens, while everyone involved is still looking somewhere else. Chapter 2 is that point.