WHITEPAPER · ORGANISATION & PEOPLE

Organise the function by horizon, not by task.

Nobody decided to stop qualifying second sources; it just stopped. A supply chain function built for a business that is doubling has to be split by the distance out each part is accountable for — because the ninety-day horizon eats the two-year one every time, and the work that protects next year is the work that never feels urgent this week.

The shape is familiar to anyone who has grown an equipment business. Procurement is one very good person. Every purchase order, every expedite, every supplier argument and every new vendor goes through them, and it works, because they are very good. Then the order book doubles. The same person is now placing the orders, chasing the late ones, arguing about the invoices, and — in the hour a week that is left — meant to be qualifying the second source for the part that has only one, negotiating the frame agreement for next year's volume, and finding out whether the contract manufacturer can actually take the ramp.

They never get to the hour, because an expedite has a customer's name on it and a qualification does not. Six months later the single-source part goes on allocation, the ramp stalls, and the post-mortem finds that nobody ever decided to stop qualifying second sources. It just stopped happening. That failure is structural: the function was built around tasks, and the short task always wins.

Split by horizon, not by task

The function I recommend — and the one we built most recently, from nothing, with the head of supply in an energy equipment business scaling into a multi-year order book: my design, her build — has three parts, and what separates them is not what they do but *how far out they are accountable for*.

Strategic sourcing owns twelve to twenty-four months: who we buy from, and why. Category strategies, the approved vendor list, second sources, frame agreements, the contract manufacturer decision, the make-or-buy question. **Procurement** owns now to ninety days: what we buy, and when. Purchase orders, expedites, receipts, invoice disputes, the daily argument with the schedule. **Logistics** owns the lifecycle of a delivery from the supplier's dock to the customer's site: how it gets here, and how we know before it is late.

Every other cut collapses. Split by category and you get three people doing the same whole job on different parts, none of them with time for the long work. Split by seniority and you get a queue, with the senior person approving what the junior one already did. Split by horizon and the parts cannot absorb one another, because each is measured against a different distance out. The person accountable for next year's supply base cannot be pulled onto this week's expedite without someone noticing that next year just lost a day.

> **The ninety-day horizon eats the two-year one every time. An expedite always feels more urgent than a qualification.**

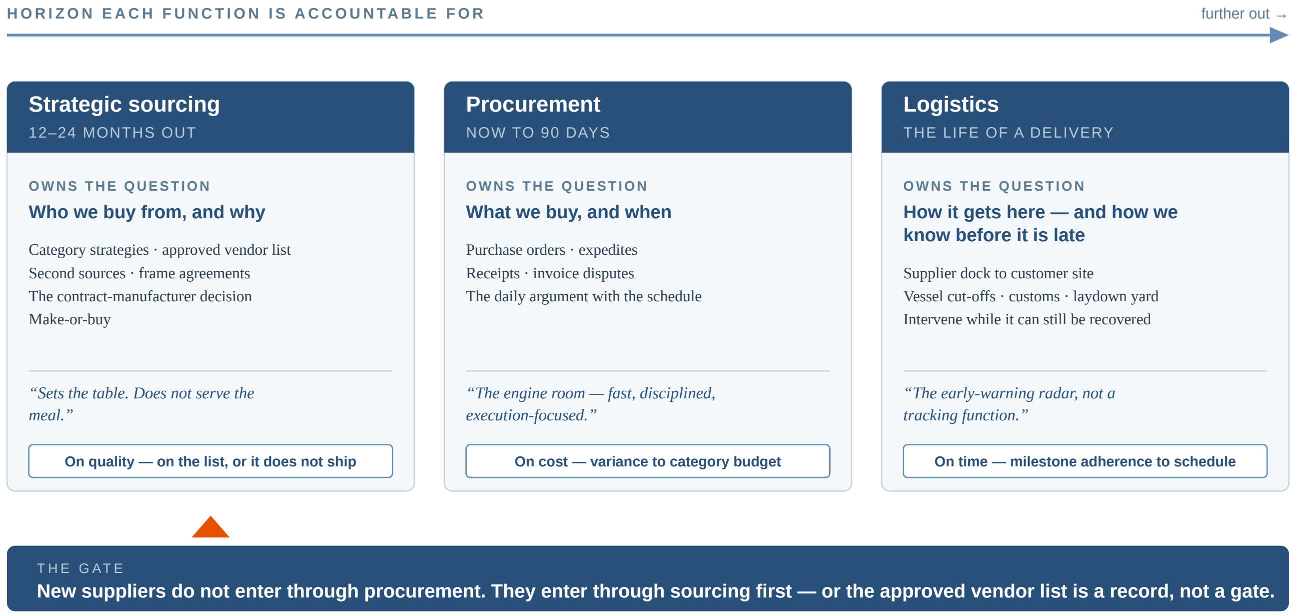

Figure 1. Split by horizon, not by task. Each function owns a different distance out, one sentence tells the business what to expect of it, and each carries one published measure. The gate sits under all three: every new supplier enters through sourcing.

This is also the argument against the thing most companies do when they are small and never undo: fold sourcing into procurement because "it is all buying". It is not all buying. One of them is a negotiation with the future and the other is a negotiation with Thursday, and if the same person holds both, Thursday wins. The horizon split is what protects the long work from the short work, and it is the only protection I have found that survives a bad quarter.

One sentence each

An org chart tells the business who reports to whom. What a counterpart in engineering or finance actually needs is one sentence telling them what to expect, and the sentences travel further than the chart. *Sourcing sets the table; it does not serve the meal. Procurement is the engine room — fast, disciplined, execution-focused.* And logistics is *the early-warning radar, not a passive tracking function.*

The logistics line is the one that earns its keep. A logistics function defined as tracking will tell you a shipment is late. A logistics function defined as radar will tell you it is *going* to be late — the vessel missed the cut-off, the customs broker has not filed, the laydown yard is not ready — while there is still something to be done about it. Same headcount, same people, different job, because the sentence set a different expectation.

The gate

One rule carries more weight than the rest of the structure put together, and it belongs on the first day of every new joiner's onboarding: **new suppliers do not enter through procurement. They enter through sourcing first.**

Without that rule an approved vendor list is a record of who has been used. With it, the list is a gate. The difference shows up the first time a project is late and an engineer has found a supplier who can ship next week: under the first regime a purchase order goes out and the company has a new, unqualified, single source with no agreement behind it; under the second, the engineer's find goes to sourcing, gets a fast-track qualification if it deserves one, and enters the list on the same terms as everyone else. Earning allocation from a supplier and owning the interface rather than the drawings both depend on this gate existing. Neither method works if the back door is open.

Three numbers, published

Each horizon gets one measure, and the three are published where the rest of the business can see them. On time: milestone adherence against the project schedule, with a threshold. On cost: variance against the budgeted cost of the category, with a threshold. On quality: binary — the supplier is on the list or the part does not ship. The thresholds are the company's to set; the pattern is three numbers, one per horizon, and no more, because a function that publishes twelve measures has published none.

They are reported weekly on one page. The version I have used most is a report card: each workstream on a row, its status in a colour, and the milestone by which it must be complete marked in the column for that month. Its virtue is that it cannot be read as anything other than what it says. Where a row has been red for three weeks the conversation is about that row, and where the deadline column has moved right the conversation is about why.

Two columns, or it is a wish list

A supply chain function does nothing alone. Every decision it makes needs something from engineering, projects, finance, quality or the scheduler, and every one of those functions needs something back. So the interface map has two columns for each counterpart — *what we need from them* and *what they need from us* — and both columns are always filled in. A one-way list of demands is a wish. Two columns is a contract, and it is the artefact that stops the "supply chain never tells us anything" conversation before it starts, because the answer is a row in a table both sides agreed.

The finding from the last build that I now carry into every one: **name the engineering functions separately.** There were five — product engineering, project engineering, research, operational technology, and the programme office that owned change control — and people routinely confused them. A map that says "Engineering" sends a question to a category and gets no decision back. A map that names the five, and what each is accountable for, was one of the highest-value pages in the whole onboarding.

> **CATEGORY COUNCILS — THE FORUM THAT IS NOT A COMMITTEE**
>
> Every function had already organised its people around roughly the same categories — the same spine, drawn independently four times. A category council makes that spine explicit across functional lines: everyone who touches a category, whichever function they sit in — the sourcing lead, the procurement lead, product and project engineering, supplier quality — against a named set of vendors. Deliberately semi-formal: no standing cadence, no minutes, no charter. Used when something is brewing in the category, not when it has already gone wrong; that single instruction is what makes it a forum rather than an escalation path. And the onboarding instruction that follows from it is the most useful thing a new joiner can do in week one: find your row, and introduce yourself to the other four names on it.

Decision rights before headcount

Structure is cheap. What makes it hold when the person who built it is away is that the decisions have been named before the people: which decisions sourcing makes alone, which procurement makes alone, which need a second signature and above what threshold. Recommend, decide, approve, by value, with a review date. If the function only works while one person with absolute authority is in the room, it is a personality, not a method, and it will not survive that person taking a holiday.

The rest is the hiring stance the method page sets out: one outcome per dedicated team, spans of five to seven, charters written for the next three roles only, and a bias to the capable person available now over the perfect one available in six months — provided the exit rule that makes fast hiring safe is written down in the same breath. The root of the whole approach is ITW's decentralised operating philosophy, which I worked inside, and it holds in businesses a fraction of that size.

How many people

The question that follows every structure is the one I cannot answer: how many people? I cannot give the number. I can teach the arithmetic, and it is the same arithmetic in every function. For procurement: sort the requisitions into difficulty classes — a repeat order against a frame agreement, a new part from an approved supplier, a new supplier altogether, an expedite. Time how long each class takes to do *well*, not fast. Multiply by the volume of each class the order book implies. Then add a service-level buffer, because every requisition arrives urgent and a team sized to average demand is late by design. The sum is the headcount, and it is defensible in front of a chief financial officer in a way that "we need two more buyers" never is — because every term in it can be argued and re-measured, and none of them is an opinion.

Where this is the wrong tool

If the business is small enough that one capable person genuinely is the function, building this early is cost without benefit: wait until the second person arrives, then build it before the third. And if the function exists and the problem is one person's performance, a structure is an expensive way to avoid a conversation you already know you need to have. The horizon split is for the business whose order book has outgrown the person, and whose second sources have quietly stopped being qualified.

That is the work I do.

WHERE THIS GOES NEXT

Who decides when you are not in the room? →

The organisation and people method — dedicated teams with named outcomes, an authority matrix, charters for the next three hires, and an operating rhythm that keeps running when its builder is away.