

Own the interface, not the drawings

Between building to print — where you own every drawing and the overhead that comes with it — and building to spec — where the supplier owns the design and, eventually, you — there is a third position most manufacturers miss. Own the boundary: form, fit and function. Let the supplier own everything inside it.

Every company that buys a complex component instead of making it lands somewhere on a spectrum, usually without having chosen the spot on purpose. At one end is build-to-print: you own the drawings, the tolerances, the materials, the revision control — every detail of what the thing is — and the supplier simply executes your design. Total control. It also means a large, permanent engineering organisation to create and maintain all that definition, a slow development cycle, and a block of fixed cost that becomes a millstone the moment the market turns down. At the other end is build-to-spec: you tell the supplier what the product must do, and they own how it does it. Fast, cheap, low-overhead, and it borrows the supplier's innovation. It also means the supplier owns the design — so you cannot second-source it, you cannot move it, and the day they decide to sell to your customer directly, you find out what lock-in actually costs.

Most manufacturers pick an end, usually by drift rather than decision, and then spend years managing its pathology. The build-to-print shop tries to get leaner and cannot, because the definition it owns requires the department it resents. The build-to-spec shop tries to claw back leverage with a cleverer contract and cannot, because you cannot second-source a design you do not own, whatever the agreement says. Both are trying to fix an architectural problem with a commercial tool, and it does not work.

The position in the middle that most people miss

There is a third place to stand, and it is the most powerful of the three because it separates the thing worth controlling from the thing that is expensive to own. Do not own the drawings. Own the interface. Define, precisely and formally, the boundary of the component — and let the supplier own everything inside that boundary. You specify where the thing connects to your system and how it must behave at that connection; the supplier specifies how to build something that meets it. This is interface-defined procurement, and

its whole promise is that it delivers what build-to-print delivers — genuine, multi-source interchangeability — without making you carry what build-to-print costs.

Form, fit, function — and the trap inside “function”

The boundary has three parts, and it pays to be exact about each. **Form** is the physical envelope: the dimensional space the component occupies, its mounting pattern, where its connectors sit. **Fit** is interface compatibility: mechanical, electrical, thermal, fluid — every way the part must mate with what surrounds it. **Function** is the performance envelope — and here is the subtlety that separates a real interface specification from a naive one. Function must be bounded in both directions. It is obvious that a component must not underperform. It is less obvious, and more important, that it must not over-perform in a way that breaks the system around it. A supplier who delivers a part that is “better” — stiffer, faster, higher-output — than the interface allows has not done you a favour; they have shipped an incompatibility. Bound the function both ways, and “interchangeable” starts to mean what it says.

A supplier cannot be better in a way that breaks the system. The performance envelope has a ceiling as well as a floor.

What you own, and what earns the label

Interface-defined procurement runs on a small set of controlled artifacts, and the discipline is in owning those rather than the internal design.

THE FOUR THINGS YOU OWN — AND NO MORE

- **The interface control document** — the single, controlled definition of the boundary.
- **The acceptance criteria** — a pass/fail specification, bounded both ways, that a third party could administer.
- **The qualification test plan** — how a supplier demonstrates it meets the boundary.
- **The approved vendor list** — the gate that says who has proven they can, held by you.

The supplier owns everything else — the internal design, the process, the manufacturing intellectual property — which is exactly the expensive part you did not want to carry. The best working model for how this holds together is the licensed compatibility badge you already know from consumer electronics, where a platform owner defines an accessory interface, licenses the right to build to it, tests against it and can revoke the licence. It is

enforced commercially, through access to the market, not through a drawing package the buyer had to author.

Define the interface, and own the gate

There is one way this goes badly wrong, and it is worth naming because it is the failure that makes people afraid of the whole idea: defining an interface and then losing control of it is worse than never defining one. The classic case is the company that specified a wildly successful hardware interface, published it openly, kept no control of the gate — and watched a clone market form around the very standard it had created and hollow out the business it was meant to protect. The lesson is not “don’t define interfaces.” It is: define the interface *and* own the gate. The interface control document is controlled, not published. The approved vendor list is a status suppliers earn against an objective standard and can lose — not a certificate they buy. The badge, if there is one, is licensed and revocable. Control of the boundary is leverage only for as long as you control who is allowed to meet it.

Define the interface and own the gate. An interface you publish but do not control is a gift to your future competitors.

You can own the interface without owning a department

The objection is always cost: surely defining and maintaining all these interfaces is its own large engineering overhead? It need not be. What you own is the boundary, not the internals — a far smaller body of work than a full print set — and you do not need a standing department to produce it. Keep a lean core design authority, and reach for time-boxed development programmes, contract and design-engineering firms, and supplier-funded non-recurring engineering and joint development where they fit. Owning the interface is a fraction of the effort of owning every drawing, and most of what it does cost can be structured as variable rather than fixed — which, in a cyclical business, is the whole difference between a capability and a liability.

Components should move, and you should know the triggers

Where a component sits is not a permanent assignment; it is a position that should change as the business does, and the error is to set it once and forget it. The usual path is a maturing one. Early on, build to spec for almost everything, because speed and learning matter more than control and the volumes are too low to justify owning anything. As you grow, move the components that carry your value onto the interface model — you own their boundary, the supplier still owns the internals. And reserve the move to full build-to-print for the handful that genuinely warrant it. The triggers that should prompt a move are specific, and worth watching for by name: volume rising to where owning the definition

pays for itself; safety or certification exposure increasing; a single supplier becoming concentrated enough to be a risk in its own right; cost pressure you cannot answer without a credible second source; or a supplier that has started to slow down, make mistakes, or look across the table at your customer. Each of those is a signal to pull a component one notch toward your control — and the interface is usually the notch to pull it to first, because it buys the control without the overhead.

None of this is a binary, and treating it as one is the original mistake. The right answer is almost always a portfolio: build to spec where speed and supplier innovation matter and the component is not where your value lives; own the interface where you need multi-source resilience and control of the boundary without the overhead of the drawings; and reserve full build-to-print for the few components so core, so safety-critical or so capacity-constrained that you must own every detail. The skill is knowing which component belongs in which bucket, and moving them deliberately as volume, risk and supplier concentration change — rather than letting each one drift to whichever end of the spectrum was convenient at the time.

And notice that none of it is a clause you insert. It is an architecture you have to design and then operate — an interface defined precisely enough to be built to blind, acceptance criteria a third party can administer, a vendor list gated on an objective standard, and the commercial nerve to revoke access when the standard is not met — inside a supply base that would frankly prefer you owned either everything or nothing. Knowing the interface is the control point is the easy half. Drawing it in the right place, and holding the gate, is the work.

That is the work I do.

WHERE THIS GOES NEXT

Who builds it, and whether the price is right →

The sourcing and contract-manufacturing method — where the interface strategy becomes a category plan, a cost model and a supplier you run. Or start a conversation: rtackconsulting.com/contact