



Engineering Before Equipment Selection

Define the Project Before the Quote

Martin Weiss

Dr Volt Inc.

Effective test systems start with business clarity and engineering discipline — not catalogs.

"We need a cyclor."

"We need a new lab."

"We need automation."

"We need to replace this system."

Statements like these show up in engineering meetings all the time. They sound like project direction, but often they are not direction at all. They are proposed solutions searching for a defined problem.

At the beginning of a project, this can feel productive. A few slides are created. Vendor websites are reviewed. Specifications are compared. Engineering teams are asked to "find a solution." Quotes arrive. Feature lists grow.

The project appears to be moving, until the full project impact becomes visible: facilities, staffing, software effort, capital cost, and operational risk.

Management begins asking different questions:

"Why are we doing this now?"

"Can we use an existing location?"

"Can we modify what we have?"

"Is this still worth doing?"

When that happens, the issue is not always poor engineering. More often, the project skipped a step. The business case was never clear enough, specific enough, or urgent enough to support the scope being explored.

That is why equipment selection should not begin with catalogs, vendor demos, or favorite platforms. It should begin with business clarity.

Engineers should not be asked to turn business uncertainty into a purchase order.

Engineering should translate business value into technical requirements, workflow definition, architecture, constraints, and measurable success criteria.

Equipment selection comes later.



When Good Ideas Fail the Business Test

A project can sound smart until it has to compete for real money, people, space, and priority.

That does not mean the idea was bad.

A new test system, lab, automation platform, or replacement system may still improve capability, capacity, labor efficiency, or long-term support risk.

The question is whether the improvement is valuable enough, urgent enough, and specific enough to justify the investment.

When that question is not answered early, engineering is left to solve an undefined value problem. The team may build feature lists, compare vendor capabilities, or pursue technically attractive options that are not anchored to a company-level priority.

Vendor input can make this worse when it arrives before the business objective and engineering problem are clearly defined.

A vendor may present a useful feature, architecture, software option, or test capability that is technically valid, but not required for the project objective. Once that idea enters the discussion, it can begin to reshape the specification.

That is how specification creep often begins. Not because the vendor is wrong. Not because the feature has no value. But because the project has not yet established which capabilities are required, which are optional, and which are distractions.

A vendor can help evaluate solutions.

A vendor should not become the source of the requirements.

Before selecting or even specifying equipment, the organization should be able to answer:

- What business outcome is this project supposed to improve?
- Why is this project important now?
- What happens if the project is delayed, reduced, or not done?
- What existing locations and resources must be considered first?
- What level of investment is justified by the expected business return?

This does not eliminate engineering judgment. It gives engineering the context needed to make better recommendations.

A Practical Flow Before Equipment Selection

Once the business reason is clear, the project needs an engineering path that prevents equipment, software platforms, or vendor features from becoming the default definition of success. The sequence below keeps the work anchored to business value while giving engineering a practical structure for decisions.

The Engineering Problem Statement

A well-defined engineering problem should answer:

- What decision must the test enable?
- What must be measured or verified?
- What level of confidence is required?
- What risks must be reduced?
- What constitutes success?

Before moving forward, this engineering problem should be checked against the business objective. The question is not only whether a system can be built, but whether the proposed engineering work still supports the business reason for the project.



Once the business need and engineering problem are clearly stated, the project can move into the company's normal engineering process. The team can evaluate possible solutions, compare technical and operational tradeoffs, and determine which path best supports the intended business outcome. New information may still emerge, and that information may require the business need or engineering problem statement to be refined. But the process now begins from stable footing: the team has a defined purpose, a defined problem, and a practical basis for making engineering decisions.



Engineering Definition Phase

Once the business need and engineering problem are clear, most engineering teams can move into a familiar engineering process. The task is to turn the problem into a workable system definition before vendor selection begins.

Define the Workflow

The workflow describes how the work is actually performed: unit preparation, measurement steps, data processing, review responsibility, automation decisions, and failure handling.

Survey the current working environment first.

Existing procedures, operator practices, maintenance issues, software limits, vendor constraints, informal workarounds, and past failures often reveal requirements the formal process misses.

The workflow should reflect how the system must work in practice. If the existing process needs to change, document that change before it becomes a hidden assumption.

Define the Requirements

Requirements describe what the system must do and how success will be measured: accuracy, resolution, timing, throughput, repeatability, safety, reporting, traceability, interfaces, operating modes, diagnostics, maintainability, and future expansion.

Define the Constraints

Constraints describe the limits the solution must work within: budget, schedule, facility limitations, existing infrastructure, available engineering resources, regulatory requirements, and long-term support expectations.

Workflow defines how the work must be performed.

Requirements define what the system must do and how well it must perform.

Constraints define the limits the solution must respect.

Final Definition Check: Separate Platform from Solution

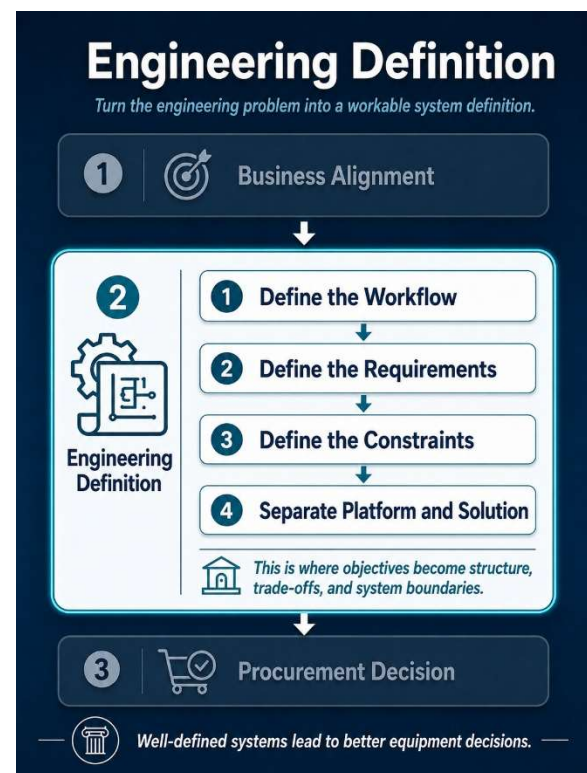
**A platform provides capability.
A solution performs the job.**

Before vendor comparison begins, engineering should separate the capabilities a platform provides from the application-specific solution the project requires.

A platform may provide measurement hardware, automation software, communications, safety infrastructure, and scalability.

The solution adds the customer-specific engineering layer: test sequences, algorithms, limits, reports, operator interfaces, diagnostics, and business rules.

Blending the platform and solution definitions can make a capable platform look like a finished system. Keeping them separate clarifies where the customer-specific engineering work begins, improves maintainability and reuse, supports future upgrades, and enables cleaner vendor evaluation.



Build the definition before evaluating options.

Workflow, requirements, and constraints keep vendor input, new ideas, and tradeoffs anchored to the project's real purpose.

What Good Selection Looks Like

Good equipment selection is rarely about identifying the “best” product.

It is about selecting the solution path that best satisfies the engineering problem, workflow, requirements, constraints, and long-term objectives.

When engineering comes first, selection becomes a structured comparison rather than a debate driven by familiarity, marketing, or existing vendors.



Engineering in Procurement

Engineers may not issue purchase orders, but they often carry responsibility for the recommendation. That recommendation is stronger when

it is grounded in the business need, engineering problem, workflow, requirements, constraints, and platform/solution boundary.

Vendor discussions may reveal useful information: a better architecture, an unexpected limitation, a simpler workflow, a support concern, or a capability the team had not considered.

That information should be folded back into the engineering definition. It should not become the new definition of the project by default.

Engineering works within constraints — including what is available, supportable, and practical within the time allowed.

Before the Purchase Order, Define the Project

Dr Volt helps engineering and management teams define test-system projects before vendor quotes drive the scope.

Start before the quote:

support@drvolt.us | www.drvolt.us