Product Flow

Measuring Execution Stability Beyond KPIs

Contents

Contents

The Problem

Today's KPI-driven management of engineering processes in mechanical engineering is largely ERP-centric. Processes are modeled around departments, cost centers, and schedules, while the product structure (BOM) is fragmented across organizational boundaries.

The product development process, however, is not department-driven but structure-based. It evolves along components, assemblies, and their temporal dependencies. When this logic is not reflected, transparency is lost — affecting real progress, risks, capacity overloads, and cross-functional interactions.

Product Flow addresses this gap by providing an independent control layer for the product development process. Rather than replacing ERP, PLM, or MES systems, it establishes transparency and control directly along the product structure and derives KPIs from real engineering work.

The Approach

Product Flow combines a B2B SaaS solution with a fundamentally rethought control model for the product development process (PDP).

Conceptually, it builds on the product-oriented management approach that was common until the 1950s, where the product structure (BOM) served as the central organizing and control element of the PDP.

With the rise of financially driven KPI systems, management focus shifted toward economic metrics and department-centric processes. As a result, the product structure faded as a controllable element — despite remaining the primary source of complexity, dependencies, and risk.

Product Flow translates this product-centric logic into a modern, data-driven software solution. The PDP is modeled along components, assemblies, and phases and linked to mathematically grounded KPIs — creating transparency and control that complements existing ERP systems without replacing them.

What Product Flow Is

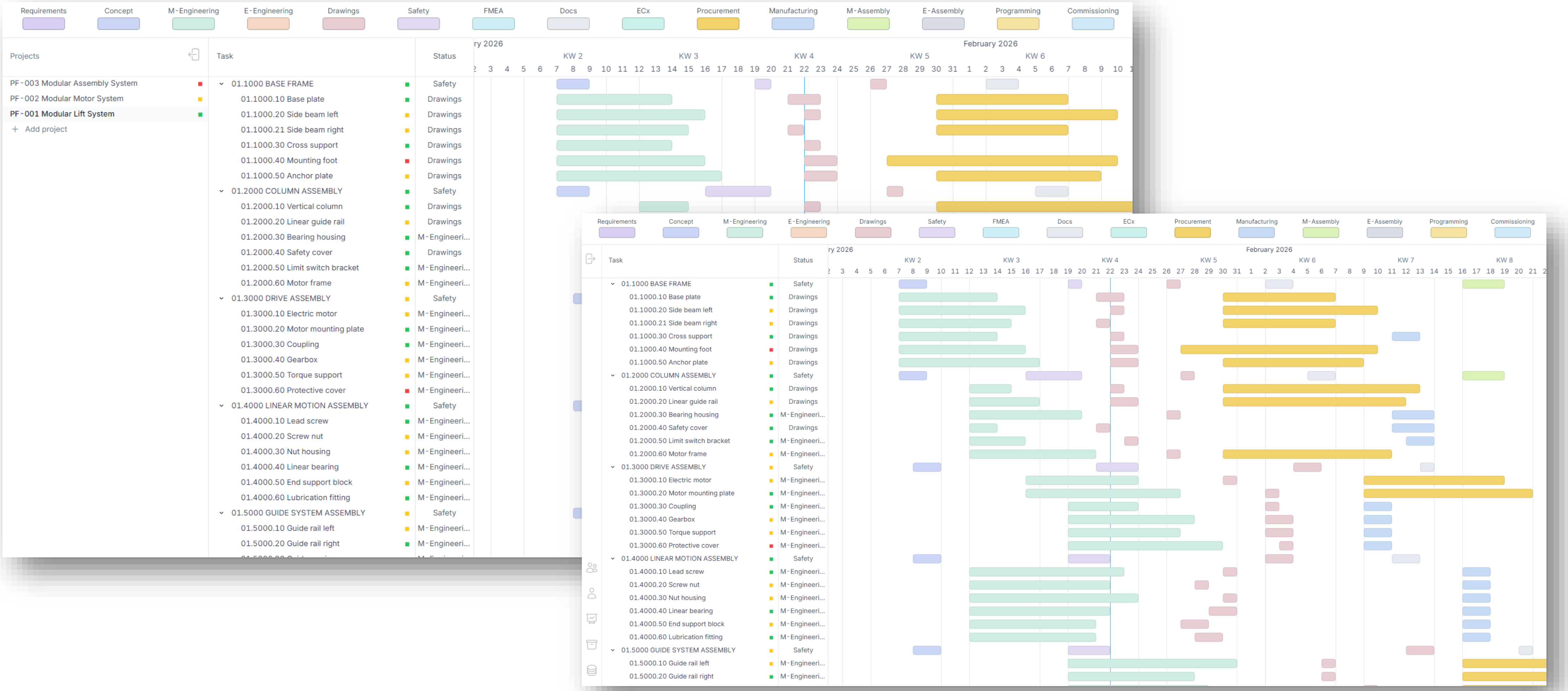
Product Flow is a digital B2B SaaS solution for the structured modeling and analysis of the product development process (PDP).

The system models the PDP consistently along the product structure. Components and assemblies (BOM) form the central organizing principle and are temporally linked to their respective process phases. Based on this structure, Product Flow processes data from real engineering work and derives mathematically defined metrics.

The Product Flow Index (PFI) quantifies structural progress and temporal stability across components, assemblies, and phases. In addition, Capacity & Structure Analysis (CSA) evaluates workload distribution, structural bottlenecks, and temporal overlaps within the PDP.

Product Flow is designed as a complementary analytics layer that integrates with existing ERP, PLM, and MES systems without replacing them.

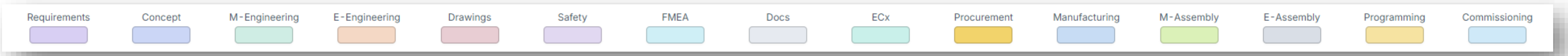
https://app.product-flow.io/



Phases

Process phases are displayed at the top of the interface and can be assigned to tasks using simple drag-and-drop interactions.

A phase is not limited to a single occurrence. The same phase can be applied multiple times to different tasks, reflecting real-world engineering workflows where similar activities recur across components and assemblies.



Requirements
Definition and consolidation of functional, technical, regulatory, and boundary requirements that form the basis of the product development process.
Concept
Development and evaluation of solution concepts, including feasibility considerations, high-level architectures, and fundamental design decisions.

M-Engineering (Mechanical Engineering)
Detailed mechanical design of components and assemblies, including geometry, materials, tolerances, and functional interfaces.
E-Engineering (Electrical Engineering)
Design of electrical systems, including wiring, control cabinets, sensors, actuators, and electrical interfaces.
Drawings
Creation and release of technical drawings required for manufacturing, assembly, and documentation.
Safety
Assessment and implementation of safety-related requirements (for example ISO 12100), including risk analysis, protective measures, and compliance with applicable safety standards.
FMEA (Failure Mode and Effects Analysis)
Systematic identification and evaluation of potential failure modes, their causes, and their effects, with the goal of risk reduction.
Docs
Preparation and maintenance of technical documentation such as manuals, specifications, declarations, and internal documentation.
ECx (Engineering Change Management)
Handling of engineering changes across the entire product development process.
ECx applies to all phases and covers change requests, evaluations, approvals, and implementation of modifications to design, documentation, or process definitions.

Procurement
Sourcing and ordering of purchased parts, components, and services, including supplier coordination and delivery tracking.
Manufacturing
Production of parts and assemblies, including internal manufacturing processes and coordination with external suppliers.
M-Assembly (Mechanical Assembly)
Mechanical assembly of components and assemblies into subsystems and final products.
E-Assembly (Electrical Assembly)
Electrical installation and wiring of components, systems, and assemblies.
Programming
Development and implementation of software, PLC logic, control programs, and parameterization required for system operation.
Commissioning
Testing, validation, and commissioning of the system, including functional tests, adjustments, and handover preparation.

Status Information

Each task has a status that reflects its current state in time — indicating what has already been completed and what is currently in progress.

Task	Status
01.1000 BASE FRAME	Safety
01.1000.10 Base plate	Drawings
01.1000.20 Side beam left	Drawings
01.1000.21 Side beam right	Drawings
01.1000.30 Cross support	Drawings
01.1000.40 Mounting foot	Drawings
01.1000.50 Anchor plate	Drawings

The status view makes it possible to see the state of every component and assembly (*Baugruppe*) at a glance. This provides immediate transparency into what has already been completed and supports an informed assessment of the overall project condition.

Status information is therefore not only a progress indicator but a key element for understanding the structural state of the project.

Gap Risk

When planning the development of assemblies (*Baugruppen*) and sub-assemblies (*Unterbaugruppen*), it is essential to anticipate risks that may disrupt the planned flow of work. Typical examples include unexpected staff unavailability (e.g. illness) or incorrect duration estimates for individual phases that require schedule extensions.

In practice, such deviations either require additional human resources or lead to project delays. In both cases, they represent a critical planning risk that must be explicitly understood and assessed.

Gap Risk describes the temporal distance to the next planned phase and therefore reflects the available buffer time. A small gap indicates that even minor deviations can immediately affect downstream activities.

Gap Risk also highlights situations where several risks occur within the same time window. When multiple gaps arise in parallel, resource constraints become critical, as project teams have limited capacity and additional tasks compete for the same resources.

Gap Risks occurring closer to the end of the project are particularly critical. At later stages, temporal buffers are reduced or no longer available, making delays increasingly difficult to compensate.

Relation to the Product Flow Index (PFI)

Each Gap Risk represents a localized planning risk at the level of components, assemblies, or phases.
The Product Flow Index (PFI) is derived from the aggregated evaluation of all individual Gap Risk indicators.

This aggregation reflects not only the existence of individual gaps, but also their temporal concentration and distribution across the product development process.

PFI specifically accounts for:

- the coincidence of multiple Gap Risks within the same time window (e.g. the same calendar week),
- the position of Gap Risks within the product development process.

When several Gap Risks occur simultaneously, overall risk increases due to limited resource availability.
Gap Risks occurring closer to project completion have a higher impact, as temporal buffers are reduced or no longer available.

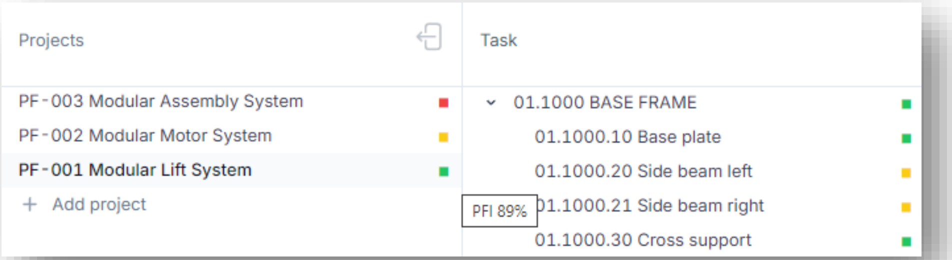
PFI therefore provides an integrated measure of progress quality and process stability.

PFI Analytical Model and Calculation Logic

The Product Flow Index (PFI) is based on a mathematically defined analytical model. It is not a heuristic score, but a structured aggregation of quantifiable risk factors derived from the product structure and timeline.

The model evaluates:

- the product structure (BOM) and process phases,
- Gap Risk indicators and available buffer times,
- temporal overlap of risks,
- and the position of risks within the project timeline.



Each Gap Risk contributes a weighted value to the index. Risks occurring in parallel and risks closer to project completion are weighted more strongly due to limited resource constraints and diminishing buffers.

The resulting PFI is a normalized value between **0 and 100**, representing the structural stability and temporal robustness of the product development process at a given point in time.

Why Classical Project KPIs Detect Risk Too Late

Classical project KPIs were originally designed for managerial reporting, not for analyzing how execution actually unfolds inside a project.
They record results after the fact and therefore always contain an inherent time delay.
In complex engineering projects, risk does not emerge suddenly.
It forms at the level of individual tasks and their transitions — between engineering, procurement, manufacturing, and assembly.

It is precisely at these transitions that the following effects arise:

- temporal gaps,
- phase overlaps,
- waiting for input data,
- blocked dependencies.

Classical KPIs do not capture these effects.
As a result, early risk detection requires a level of measurement below KPIs — at the task level, where gap risk indicators can be applied to reveal emerging instability long before deadlines are missed.

Aggregation and Time: From Gap Risk Indicators to Systemic Project Risk

Project KPIs aggregate execution at the level of phases and the overall project.

In doing so, local gap risk indicators that arise at the level of individual tasks cancel each other out and disappear in the aggregated view.

However, it is precisely the sequence and temporal distribution of such gaps that form the project's systemic risk. Not an isolated delay, but the accumulation and mutual overlap of multiple gaps is what makes execution unstable.

This creates the need for a second level of analysis an indicator that does not merely sum risks, but takes into account:

- the position of tasks in time,
- the coincidence of critical phases,
- the density and concentration of risks.

Such an indicator enables the transition from local gap risk indicators to an integrated assessment — PFI (Project Flow Index) — which reflects how evenly and sustainably execution unfolds across the project as a whole.

Early Risk Visibility: From Reaction to Management

When gap risk indicators are identified at the task level, risk becomes visible before it appears in KPIs.
Management gains the ability to respond to causes, not consequences.
PFI, in turn, makes it possible to see how local gaps and overlaps transform into project-level execution risk, long before this risk materializes as missed deadlines or resource overruns.

In this case, management ceases to be reactive.

Decisions are made based on:

- the structure of execution,
- task dynamics,
- the temporal distribution of risk.

Classical KPIs do not lose their relevance.
They remain essential tools for reporting and outcome control.
However, managing execution risk requires an additional layer of visibility — from gap risk indicators at the task level to PFI at the project level.

Why the Same Total Risk Can Result in Different PFI Values

At first glance, it may seem that project risk is determined by the sum of individual task risks.
If the total number of issues is the same, the overall level of risk should be comparable.
In practice, this assumption is incorrect.

Execution risk is determined not only by magnitude, but by the distribution of risks over time and within the project structure.

Two projects may have the same total load of gap risk indicators, yet differ fundamentally in execution stability.
If risks are evenly distributed and occur in different time windows, the project retains its ability to compensate:
teams have time to react, dependencies are not overloaded, and local issues do not amplify one another.
If the same risks are concentrated in time or overlap with critical phases, an amplification effect occurs:
tasks begin to block each other, attention and resources compete, and local delays quickly escalate into systemic instability.
PFI captures precisely this difference.
It reflects not merely the presence of risks, but the nature of their distribution and mutual interaction within the project's execution flow.

PFI as an Indicator of Execution Stability, Not Deadlines

PFI does not answer the question: “Will we meet a specific date?”
It answers a different, more fundamental question: **“How sustainably is project execution organized over time?”**

Deadlines are an outcome. Execution stability is the cause.
A project with high execution stability may temporarily deviate from the plan and still remain manageable.
A project with low execution stability may formally stay on schedule while existing in a state of latent risk, where any deviation triggers a chain reaction.

PFI reveals:
- how dense and overloaded execution is,
- where risks are concentrated,
- whether the project is losing its ability to self-compensate.

Thus, PFI does not replace project KPIs.
It complements them by shifting the focus from date control to the assessment of structural execution stability.
This is why PFI makes risk visible earlier than it appears in schedules, budgets, or status reports.

Why Projects That Are “On Time” Still Fail

Formal adherence to deadlines does not imply execution stability.
Many projects appear “green” right up to the moment they suddenly lose manageability.
This happens because classical plans and KPIs record the fact of completion,
but do not reflect the state of the execution flow through which that result is achieved.

A project may remain on schedule through:

- temporary compensations,
- overloading key specialists,
- shifting risks to later phases,
- implicit assumptions of “we’ll deal with it later.”

From the outside, such a project appears stable. Internally, however, it gradually loses its ability to respond to deviations. When additional risk eventually materializes — an unexpected change, a delivery delay, a technical issue — the project no longer has any reserve of flexibility. Failure does not occur because the project was “poorly planned,” but because its execution stability was exhausted long before the deadline.
This is why projects often fail *on time* — they reach the critical point precisely when compensatory mechanisms stop working.

PFI → Management Decisions: What to Do When PFI Increases

An increase in PFI is not a signal for panic and not an indicator of failure. It is an early sign that project execution is becoming **structurally unstable**.
Unlike KPIs, PFI does not require immediate "schedule correction." It creates space for management decisions **before escalation occurs**.

When PFI increases, the following types of decisions become possible:

- **Reallocating attention rather than resources** (removing parallel tasks from critical nodes)
- **Separating phases in time** (reducing overlaps, even without changing final deadlines)
- **Reducing execution density** (deliberate deceleration instead of uncontrolled acceleration)
- **Prioritizing by stability rather than urgency** (identifying tasks that reduce overall flow risk)

What matters is that these decisions are made at the level of execution structure, not through pressure on teams or adjustments to reporting. PFI does not prescribe what exactly must be done.

It reveals where the project is losing its ability to self-compensate. This is what makes PFI a management instrument: it shifts the conversation from *"Will we make it?"* to *"How sustainably are we actually moving?"*

From Observation to Management: How Measurement Changes Behavior

Measurement does not manage a project by itself. But it defines **which actions are considered rational**. When only deadlines and budgets are measured, behavior shifts toward:

- maintaining formal status,
- last-minute compensations,
- hiding instability until the next report.

Attention is directed at outcomes, not at the conditions under which those outcomes are produced. When execution structure becomes visible — task distribution, phase density, temporal risk concentration — the logic of management changes.

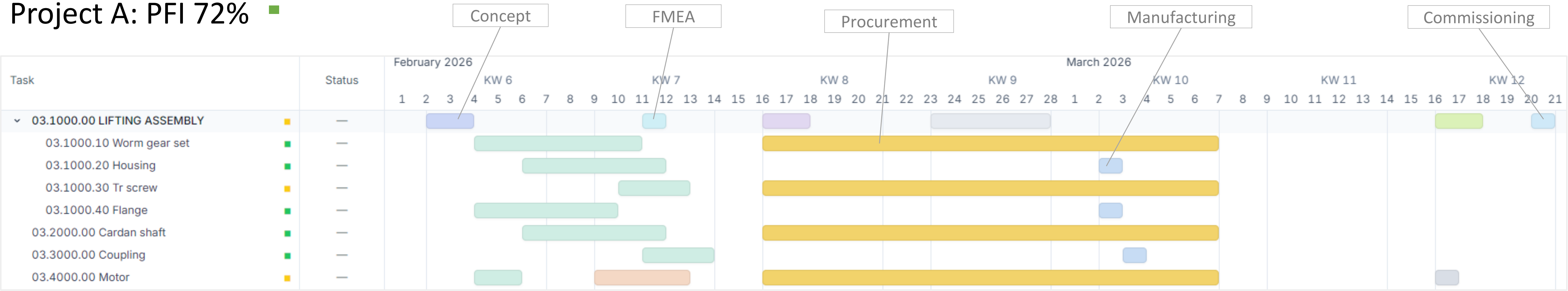
Decisions are made:

- earlier,
- based on causes rather than consequences,
- without pressure or escalation.

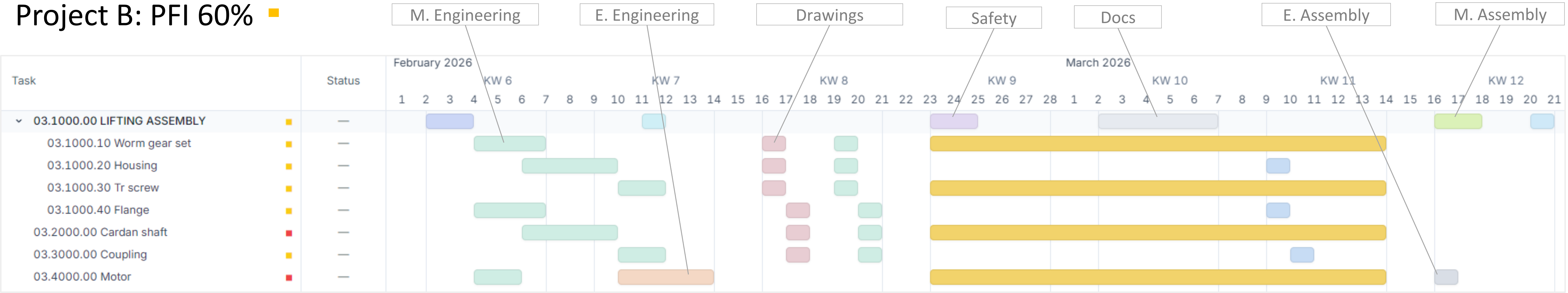
PFI does not add another KPI. It shifts management focus from **controlling results** to **maintaining execution stability**. This is what turns observation into management.

Project A vs Project B: Same Deadlines, Different PFI

Project A: PFI 72%



Project B: PFI 60%



Both projects were defined under identical baseline conditions:
the same overall project duration and the same planned duration of individual phases.

In **Project B**, drafting work was performed by dedicated drafters, while in **Project A**, engineers were responsible for both 3D modeling and 2D drawings.
The involvement of drafters in Project B was planned in advance and formally scheduled — which is effective under stable interfaces and frozen inputs.

The comparison clearly shows that in **Project B**, a delay in the delivery of purchased components inevitably leads to a missed project deadline.
Even with proper planning, the role separation reduces flexibility once upstream uncertainty occurs.
The execution structure provides no temporal buffer to absorb disruptions.

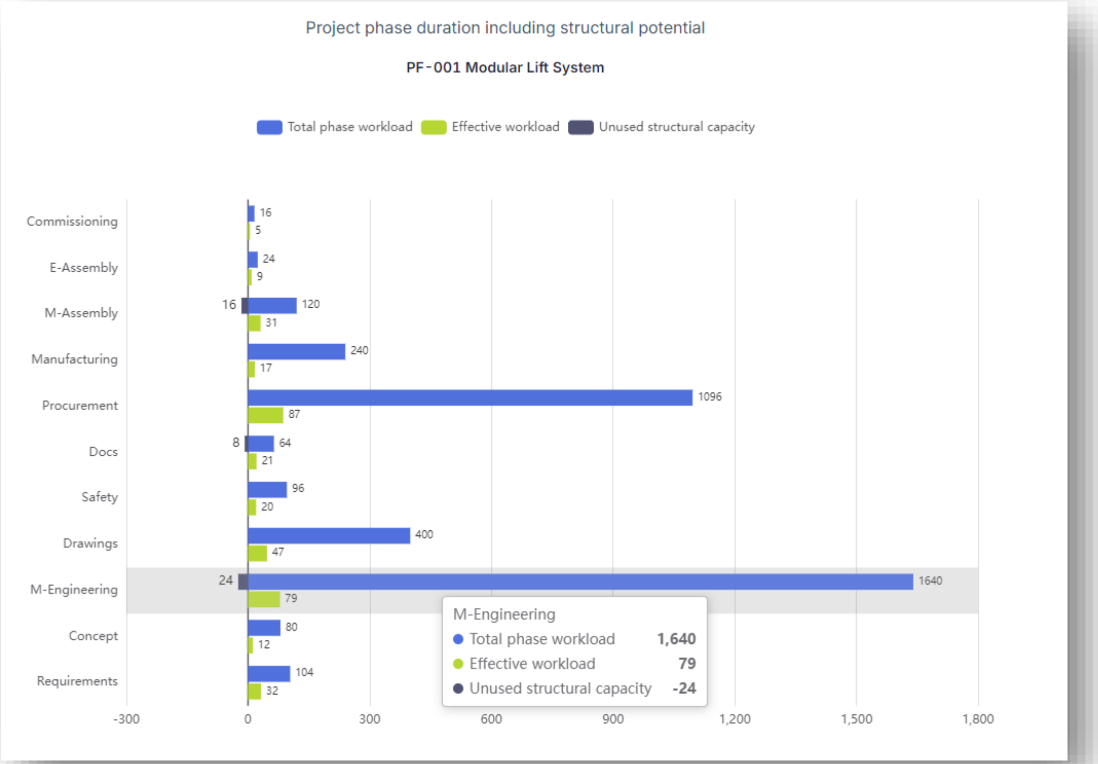
In **Project A**, a time buffer is deliberately built into the execution flow.
If the delivery of components is delayed or engineers decide to introduce design changes, the project remains manageable because execution can adapt without crossing role boundaries.

If component delivery proceeds as planned, this buffer can be used proactively —
allowing the project to be completed and delivered to the customer **up to one week earlier**.

Structural Capacity Analysis (CSA)

While PFI describes execution stability, CSA explains how capacity is structurally consumed.

Structural Capacity Analysis (CSA) evaluates how engineering and production capacity is structurally utilized across the phases of a project:



1. Total Phase Workload - This represents the total amount of work assigned to a given phase, aggregated from all related tasks, components, and assemblies.

2. Effective Workload - The effective workload reflects how much of the total phase workload can realistically be executed within the available time window.

In this calculation, CSA explicitly considers that:

- a team member cannot work more than **8 hours per day**,
- tasks scheduled on the same day are distributed accordingly across available working time.

This ensures that workload calculations are grounded in realistic human capacity rather than theoretical availability.

3. Unused Structural Capacity - Unused structural capacity indicates the difference between the available execution window and the effectively utilized workload.

A positive value reflects unused potential, while a negative value indicates structural overload — meaning that the planned workload exceeds what can realistically be executed within the phase.

Comparative Interpretation

By comparing these three CSA parameters across similar or structurally comparable projects, it becomes possible to perform a comparative analysis of team efficiency and execution effectiveness.

If projects with comparable scope and structure require significantly different time resources, this indicates that additional factors may be influencing performance. Such deviations often point to underlying organizational, coordination, or management-related issues rather than purely technical differences.

CSA therefore provides a quantitative basis for identifying where deeper analysis of planning, coordination, or leadership practices may be required.