Process Flow Visibility

A practical method for making work visible — where time is spent, where delays occur, where rework hides, and where risk accumulates. It works the same way on a vulnerability, a project, an access request, a recovery, an audit finding or a customer order; only the cost of the waiting changes.

Before you buy more people, technology, or services — do you know how effectively your existing resources are being converted into outcomes? PFV shows where an investment would actually improve capacity, flow, throughput or exposure — and where it would not.

Where this came from

Throughout my career I was responsible for increasingly complex technologies, services and organisations, and I learned that **I did not need to be the deepest technical expert in every area to lead effectively. What I needed was visibility.** Process mapping became my way of turning complexity into understanding and uncertainty into measurable facts. Process mapping is the data collection mechanism; Process Flow Visibility is the visibility created from that data. Four free browser tools do the counting, from a fifteen-second napkin calculation to full worksheets, with 55 worked samples built in.

What I keep finding

Most organisations can say how hard their people are working and very few can say how long the work waits. When I map a recurring process end to end, **one to five percent of the elapsed time is usually somebody doing something**; the rest is the item sitting. That is not a failure of effort — the work lives *between* people rather than with them — and it stays invisible until somebody counts it.

What forty-two worked examples suggested

Maturity level	Total lead time	Removed	Cumulative	Median PCE
2 — Managed (ad hoc)	809 d	—	—	2.8%
3 — Defined (standard work)	497 d	312 d	39%	4.9%
4 — Quantitatively managed	324 d	173 d	60%	8.3%
5 — Optimizing (self-improving)	**199 d**	**125 d**	**75%**	**10.3%**

Every one improved at every pass — though the figures are modelled, so the pattern is the useful part, not the decimals. Personal-life processes are excluded; they behave differently enough to blur the picture.

The three things worth knowing

1 Most of the time, nobody is working. One to five percent of the elapsed time is somebody actually doing something; the rest is the item sitting, waiting for a person, an approval, a system, or a decision. Not because anyone is idle — they are busy on something else — which is why adding people so rarely helps.

2 Your team is not slow — worth saying out loud in front of them. The work does not spend its life *with* people, it spends it *between* them, so everyone could work twice as fast and the number would barely move.

3 Not every wait you remove gives you capacity back. Removing a wait returns one of three things: **capacity**, when effort is eliminated and work stops being done twice; **flow**, when each item simply moves faster; or **less exposure**, when the condition stays open a shorter time. Only the first lets the same people finish more. So ask whether shortening a wait finishes *this item* sooner, or lets the system finish *more items*.

And if AI did the work instead?

	Total lead time	Cut
Replace the humans with AI, same process	255 d	68%
Keep the humans, fix the flow	**199 d**	**75%**
Both	**54 d**	**93%**

In this modelled set AI removed 80% of the work but only 66% of the waiting, and lead time is made of waiting. On 22 of the 42, fixing the flow with the same people came out ahead. Treat that as a reason to map before deciding, not as a prediction.

Where it does not apply

PFV decomposes flow, and not all delay is flow: in one worked case, 54 of 63 days of intruder dwell time were not a queue at all — nothing was looking, found only by subtraction. Being clear about where the method stops is what makes the rest of it worth trusting.

What an engagement looks like

1. Start how many items are open, how many finish in a period. Divide. Does that cycle time work for the business? If it does, stop.
2. Map one recurring process. Per step: the task, who performs it, hands-on work, waiting, whether that time is owned by an external team, and where work is sent back to an earlie
3. See it as one timeline drawn to scale — green for work, red for waiting you control, yellow for work or waiting owned outside your organisation.
4. Classify the waits — internal, external, required, or rework related.
5. Diagnose where the largest controllable exposure sits, and why the work waits there.
6. Improve one thing. Then measure the same process again.

Step 2 turns a deliverable into evidence; step 5 is the only way anyone learns whether the recommendation landed. You leave with the map as a file, the process drawn to scale, and a diagnosis naming which waits are internal, external, required or rework.

Evidence status. The 42 processes are realistic samples modelled from industry data rather than audited case studies, and the improvement and AI figures come from rule-based models calibrated against the one process built at four levels by hand. Their value is the pattern, not the numbers, and I would not present them as proof. Every starting figure is reproducible by loading the sample in PFV. Field trials on real timestamps are the open gap; results will be published whichever way they land. CMMI® is a registered trademark of ISACA; this work is independent and is not affiliated with, endorsed by, or appraised under CMMI Institute or ISACA.