OPERATIONAL EXCELLENCE · FRAMEWORK

Process Mapping & Bottleneck Framework

A practical way to find where an operation actually loses time and money — and what to fix first.

Most operations don't fail everywhere at once. They lose time and money at one or two specific points — the constraint — while everyone spreads effort evenly across the whole thing. This framework is about finding that point before you spend a penny fixing anything.

The five steps

1 **Map the flow as it really is.** Not the official process — the actual one. Walk it end to end, step by step, and write down every hand-off, wait, and rework loop. Talk to the people doing the work; they know where it breaks. Draw it simply: boxes for steps, arrows for flow.

2 **Measure each step.** For every step, capture two numbers: how long the work takes, and how long it waits before it starts. In most operations the waiting dwarfs the working — that gap is where your opportunity hides.

3 **Find the constraint.** The bottleneck is the step that sets the pace for the whole flow — work piles up in front of it and starves everything after it. There is usually exactly one that matters most. Improving anything else is motion without progress.

4 **Attack the bottleneck — in order.** First, get more out of the constraint as it stands (remove interruptions, stop it doing work that isn't needed). Only then consider adding capacity or automating. Squeeze before you spend.

5 **Standardise, then re-check.** Lock in the improved way of working so it doesn't drift back. Then re-map — because the moment you relieve one bottleneck, a new one appears somewhere else. Improvement is a loop, not a project.

THE ONE IDEA TO KEEP

An hour saved at the bottleneck is an hour gained for the entire operation. An hour saved anywhere else is an hour gained nowhere. Spend your attention accordingly.

A 60-second worked example

An after-sales team took, on average, nine days to close a customer issue. Mapping the flow showed the actual hands-on work was under four hours — the other eight-and-a-half days were the ticket *waiting*: for triage, for a specialist, for parts. The constraint wasn't effort; it was the triage step, where everything queued.

The fix wasn't more engineers. It was resolving more issues at first contact so they never entered the queue — which cut the backlog at the constraint and pulled the whole cycle time down. Same team, same tools, a fraction of the time.

Before you start — a checklist

- [ ] Have I mapped the process as it *actually* runs, not as the SOP describes it?
- [ ] Do I have waiting time as well as working time for each step?
- [ ] Can I point to the single step where work piles up?
- [ ] Have I tried to get more from the constraint before proposing to spend on it?
- [ ] Do I have a way to stop the improvement quietly reverting?