Work Management / CMMS

Work Management Alignment

Why your CMMS isn't delivering your strategy

A SEAMPAS Perspective · April 2026 · *Approx. 10-minute read*

There is a promise embedded in the purchase of any modern CMMS: that once this is implemented, the maintenance actions your engineers have determined to be necessary will be the ones that get planned, scheduled, executed, and recorded. Strategy and execution will be joined up. The loop will be closed.

In the majority of the organisations we have worked with, the loop is open. Wide open, and the CMMS, far from being what closed it, is often the reason it stays that way. The software is fine; the discipline around it is not. The result is a work management system that produces impressive reports, generates thousands of work orders a month, and delivers almost none of the strategic value its proponents predicted when it was installed.

This piece is about why that happens, how to see it happening in your own organisation, and what it takes to close the loop. We are less interested here in the features of any particular CMMS as they are all broadly capable, and more interested in the four structural practices that, in our observation, separate a work management system that delivers strategy from one that merely records activity.

The promise, and the reality

The promise of a CMMS is a closed loop between strategy and execution. The strategy lives in the FMEAs, the criticality assessments, the maintenance tactics, the spares policies. The execution happens on the floor people with tools, at specific assets, on specific shifts, doing specific things. The CMMS is meant to be the connective tissue: the thing that translates strategic intent into scheduled work, tracks that work through completion, and feeds the results back to strategy.

When the loop is closed, an organisation can answer a simple question: *why are we doing this particular PM task on this particular asset?* The answer traces back through the job plan to a maintenance tactic, from the tactic to a failure mode in an FMEA, and from the FMEA to a criticality rating that justifies the attention. Every task is justifiable, every work order is traceable, and the aggregate data flowing out of the system is in principle an informed commentary on how well the strategy is working.

When the loop is open, that question has no good answer. The PM task is being done because it was there last year, or because a vendor recommended it once, or because someone's manager added it in 2014 after an incident, or because it was imported from the previous plant in the previous merger. The work is happening; the justification is not. The data flowing out of the CMMS is neither informed by strategy nor informing it. It is just data.

The four practices that most reliably open that loop, and keep it open, are the subject of what follows.

The hierarchy you didn't know you had

The asset hierarchy is the spine of the CMMS. If it's broken, everything built on it - job plans, work orders, cost rollups, reliability reports, spares integration, is broken too, in ways that will not be obvious until you try to trust the output.

In our experience, the hierarchy is the first thing to rot and the last thing to get fixed. It is typically built early in the CMMS implementation, by someone who may no longer be with the organisation, using information that may no longer reflect the plant. It is then amended continuously, a motor added here, a skid merged there, an area renamed after a reorganisation with little or no governance. After a few years, a single piece of equipment might appear under two or three different parent hierarchies, or be orphaned entirely, or share a tag with something in a different area that was never deregistered.

The symptom to look for is simple. Ask your CMMS to produce a cost rollup for a specific functional area and ask separately for the same rollup by equipment type. If the two totals are materially different, your hierarchy is fragmented. If they're only different by a small margin, the fragmentation might still be large where small discrepancies often mask large reconciliations further down.

Fixing the hierarchy is not glamorous work. It is slow, it is boring, and it produces no visible output until the whole job is done. For these reasons it is almost always deferred, and for the same reasons it should almost always be the first item on any serious work management improvement effort. Everything else built on a broken hierarchy inherits the break.

Every job plan needs a parent

Most organisations we've worked with have between 500 and 5,000 active job plans in their CMMS. In almost every case, nobody owns them individually. They were created over a decade or more, by many different people, for many different reasons, and they have never been subjected to structured review.

The consequences are predictable. We have seen six-monthly lubrication tasks on assets that have no lubrication points. We have seen condition monitoring tasks that reference instrumentation removed five years ago. We have seen identical tasks duplicated across three job plans against the same asset, producing three separate work orders each cycle. Job plans that reference regulatory requirements no longer current in the jurisdiction, and job plans that reference vendor recommendations from vendors that no longer exist.

None of this happens because people are incompetent. It happens because a job plan, once created, has no natural expiry mechanism and no natural owner. The person who created it moves on; the job plan doesn't. And so the population grows, the signal-to-noise ratio decays, and the PM programme becomes a geological record of past decisions rather than a coherent expression of current strategy.

The response is a simple governance discipline: every job plan has a named owner (a role, not an individual), a review cycle, a traceable link to the FMEA or tactic that justifies it, and a retirement trigger when that link can no longer be demonstrated. We have written about this at length in a companion Methodology Paper on FMEA-to-AMT correlation; the headline point is that without such governance, job plan populations always grow, and their quality always decays.

> *Schedule compliance at 95% is sometimes an achievement and sometimes a measurement artefact. The difference depends entirely on how the organisation handles the difference between the work it was supposed to do and the work it actually did.*

When reactive work eats the schedule

A well-run maintenance organisation does most of its work on plan. A typical modern target is something like 75–80% planned, with the balance reactive. The KPI that measures this, usually schedule compliance, is one of the most misunderstood and most gamed metrics in the entire work management canon.

Here is what happens in practice. The week begins with a published schedule: so many PM tasks, so many planned corrective tasks, a small allowance for reactive demand. On Tuesday, something breaks. The crew is pulled off a planned inspection to address it. On Wednesday, something else breaks. On Thursday, three more things break. By Friday, the crew has done very little of the planned work and a great deal of reactive work. At the end of the week, the planned tasks that didn't get done are either rescheduled (if the system is well-governed) or quietly dropped (if it is not). Schedule compliance for the week: 95%, because it was calculated against the tasks that were *attempted* rather than against the tasks that were *planned.*

This is not a CMMS problem. The CMMS will calculate whatever definition the organisation configures. It's a discipline problem: the organisation has decided, implicitly, reactive work has priority over preventive, the preventive schedule is advisory, and compliance is a narrative to be managed rather than a signal to be read.

The counter-practice is simple to describe, difficult to enforce. Protect the preventive schedule. Distinguish formally between the work that must happen this week (PM programme) and the work that can legitimately slip (minor inspections, condition monitoring tasks with long P–F intervals). Report compliance honestly, against the original plan. When the reactive demand is so high that the planned schedule cannot be maintained, that is the *signal* - not something to be managed around, but information to be responded to.

The data that goes in and never comes out

Every work order completed in a CMMS generates a record: what was done, by whom, when, for how long, at what cost, with what findings. Over a year, this data accumulates into an extraordinary operational record, one of the richest datasets any industrial organisation produces about its own performance.

In practice, almost nobody reads it. Reports produced tend to be aggregate and lagging, MTBF by equipment class, cost by area, schedule compliance by crew, all of which have their uses but none of which tell you what your failure modes actually are, how often they recur, and whether your maintenance strategy is addressing them. The fields that tell you that are, failure mode classification, root cause, observed condition are typically either not populated or populated with free text that cannot be analysed in aggregate.

This is the fourth and most consequential failure of the loop. The feedback arm, execution data informing strategy review, is broken. And because it is broken, the FMEA library and the plant drift apart a little further every year. New failure modes emerge and are never captured. Addressed failure modes stop occurring and remain in the programme anyway. The strategy gets further from reality with each revision, and the organisation loses the ability to know it.

Fixing this is more cultural than technical. Someone needs to be responsible for reading the work order data every month, a named role, with time allocated, with a defined output. The output is not a dashboard. The output is a list of observed failure modes, cross-referenced to the FMEA library, with proposed strategy adjustments. That list goes to a governance forum. The governance forum decides what to change. The changes propagate to the job plans. And the loop closes.

Closing the loop

The four practices are not independent of one another. A fragmented hierarchy makes job plan governance harder. Weak job plan governance makes schedule compliance meaningless. Poor schedule discipline starves the feedback loop of reliable data. A broken feedback loop allows the hierarchy and job plans to drift further. They reinforce each other, for better or worse.

This means that attempting to fix one in isolation tends to disappoint. Organisations that try to "improve schedule compliance" without first fixing the job plan population end up gaming the metric. Organisations that try to "analyse the work order data" without first fixing the failure-mode coding end up applying sophisticated tools to unreliable data. The practices close the loop together - or not at all.

The order we've seen work is roughly: **hierarchy first**, job plan governance second, schedule discipline third, feedback loop fourth and then all four as an integrated set, with each feeding the others. It is not quick, and it does not deliver a dramatic before/after graph. It is slow, unglamorous, and discipline heavy. What it *does* deliver is a maintenance programme that is actually traceable to strategy, and a CMMS that actually closes the loop it was purchased to close.

A note on discipline versus technology

In almost every organisation we've worked with, the explanation offered for the open loop is "we need a better CMMS." It is almost never the right answer. The CMMS in question is usually capable of everything being asked of it; what is missing is the discipline around how it is used. This is perhaps the most uncomfortable message a maintenance improvement exercise can deliver. That the solution is not a purchase order but a change in practice and it is also the most consistent finding we have.

Closing the loop is, in the end, not a technology problem. It is a question of whether an organisation wants its maintenance strategy to be a document that sits in a SharePoint site or a living practice that flows through its work management system and out into the plant. The difference between those two states is not the CMMS. It is the discipline.

About this Perspective

SEAMPAS Perspectives are short, opinion-led pieces drawn from SEAMPAS practice across mining, minerals processing, utilities, and heavy rail. They are intended to provoke thought rather than to prescribe method, and they sit alongside the more structured SEAMPAS Technical Notes and Methodology Papers in the SEAMPAS publications series. Reactions, disagreements, and counter-examples are welcome and encouraged.

Related SEAMPAS publications. *Technical Note 01: A Practitioner's Guide to FMEA in Mining Environments; Technical Note 03: Maintenance Criticality Frameworks; Technical Note 05: Condition Monitoring Selection; Methodology Paper 02: FMEA-to-AMT Alignment.*

Correspondence. john@seampas.com · SEAMPAS Pty Ltd, Roleystone, Western Australia.