# Chapter 6. Solvable problems

## Introduction

We commonly describe work by occupation or activity: carpenter, engineer, doctor, supervisor, chief executive or parent. Those labels tell us what social role a person occupies, but they do not tell us the level of problem being solved. This chapter uses Elliott Jaques' work on stratified systems to describe problem solving more scientifically: movement from a present state A towards an intended state Z, using discretion, within constraints, over a time span, with consequences for success or failure. The chapter then distinguishes between problems that can be solved and paradoxes that cannot be finally solved but can only be held.

Levels of problem solving give work a more useful description than occupation, title or effort. They show what kind of movement from A to Z is required, how much discretion is available, what constraints apply, how long consequences take to emerge and what level of judgement the work demands.

At lower and middle levels, problems are primarily solvable. The pathway may be difficult and uncertain, but a desired state can be stated and feedback can show whether the action worked. Maintenance, as an example, makes this progression visible: the work rises from repairing an object after failure to sensing deterioration, eliminating recurring causes, predicting disruption and redesigning the architecture that generates failure.

At Levels VI and VII, work becomes **paradoxical**. The task is no longer to discover one final answer, because the system must preserve multiple competing necessities. Freedom and order, competition and cooperation, truth and harmony, present and future, national and global cannot be solved away. They must be held, monitored and repeatedly rebalanced.

AI can greatly strengthen bounded analysis and control, but at the **paradox levels it should illuminate the field** rather than claim the authority to settle it.

The levels of problem solving show a clear change in the character of work. Many problems have an intended result, a pathway that can be tested and feedback that can confirm completion. At the higher levels, the work increasingly involves values and needs that remain necessary together: freedom and order, competition and cooperation, truth and harmony, present needs and future responsibility.

Why read this chapter? Because the aim comes first. If the aim is wrong, unclear or set at the wrong level, we can solve the wrong problem brilliantly.

This chapter gives us a practical way to locate the aim, locate the problem, locate the problem solver and then choose the right kind of work.

## How this chapter unfolds

**1. Putting work into a scientific description of problem solving** - Work is movement from a present state A to an intended state Z, using judgement within limits, over time, with consequences.

**2. Problem solving changes by level** - As work rises in level, aims widen, time horizons lengthen, information becomes less certain and more parts of the system must be held together.

**3. Ten step process to solve levelled problems** - A ten-step process first locates the condition, the problem and the problem solver, then moves through definition, analysis, choice, action and learning.

**4. Solvable problems** - A solvable problem has an aim that can be described, evidence that can be gathered and a result that can be tested.

**5. Maintenance as a levelled solvable problem** - Maintenance shows how the same work changes as the aim moves from fixing a failed part to redesigning the system that creates failure.

**6. When problem solving becomes paradox-holding** - At the highest levels, work moves from solving one bounded problem to holding necessary but competing goods in balance.

**7. What this means for humans** - The earlier chapters and this one together show why humans must get the aim right, locate the problem at the right level and keep truth, judgement and correction connected.

**8. What this means for AI** - AI can solve and support many bounded problems, but humans must still set the aims, choose the level and legitimately settle higher-level paradoxes.

## 1. Putting work into a scientific description of problem solving

The word work is slippery. We tend to describe work in terms of what is done — carpenter, supervisor, engineer, doctor, customer service — and this everyday language is recognisable but not helpful for analysis. Analysis requires normative description and normative measurement.

I was fortunate to work with Elliott Jaques who developed Stratified Systems Theory (SST) which looks at work with scientific rigour and Sir Roderick Carnegie who applied SST at Rio Tinto. Elliott Jaques gave work a normative structure by defining it as **movement from state A to state Z, using discretion**, within **design and operational constraints**, over a **defined time-span**, with **consequences** for success or failure. That definition allows work to be measured and levelled. Each element of the normative structure can be defined:

**Path from A to Z** - Work is movement from a present state, A, to an intended state, Z. The intended path to state Z is itself fixed by the operating constraints — the outputs, standards and conditions that define what reaching Z actually means. The path is the route taken across that gap: the sequence of interventions, tests and corrections by which the present condition is changed into the intended one. Because actions are sequential and reality answers back, the path is rarely a straight line; it is chosen, attempted and revised as feedback arrives.

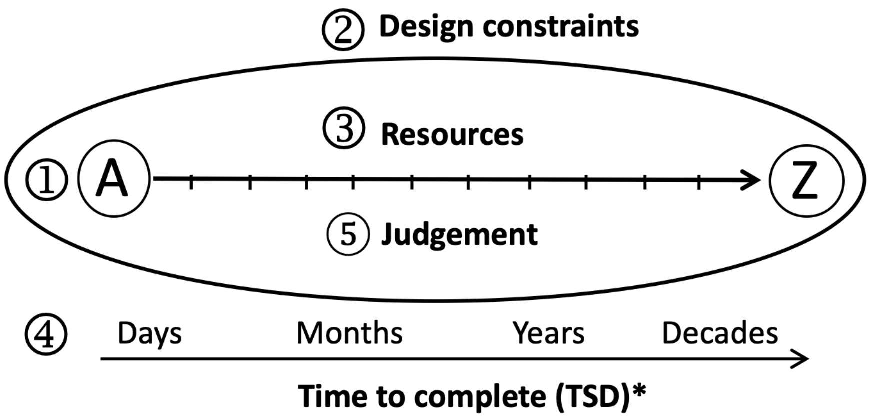
**Design constraints** - The non-negotiable boundaries within which the work must be done: the standards of safety, quality, law, brand, fiduciary duty and accepted practice that define what must never occur. Set from the level responsible for the larger whole, they establish the operating envelope and cannot be traded away in the pursuit of Z.

**Resources** - The means that can be mobilised to move from A to Z — principally people, money and equipment, and also time, information and capacity. Resources are finite, which is part of what makes the work a problem: the person must allocate and sequence limited means, and the number and level of people who can be brought to bear is itself a constraint on what is achievable.

**Time to complete** - The period required to reach Z, which lengthens as the task becomes more complex, more contested and more significant. It is driven by the sequencing of actions, the resolution of resource and interest conflicts, and the consequences that must unfold before completion can be confirmed. The longer this span, the higher the level of the work.

**Use of judgement** - The discretion exercised in selecting and revising the path within the constraints. Where information is incomplete and the answer cannot be specified in advance, the worker must interpret, decide and bear accountability for the consequences. The amount and quality of judgement required is the truest measure of the level of the work.

**Normative description of Work. Exhibit 6.1**




1. **Start** A → **Aim** Z
2. **Design** Constraints
3. **Resources** – people, $, equipment
4. **Time** to complete task, dependent on conflicts and significance of complexity
5. Use of **judgement** as information changes

*TSD – time span of discretion

## 2. Problem solving changes by level

Levels of work can be readily seen in the standard organisational structure. As work rises in level, the problem changes. The worker must integrate more conflicting functions, larger spaces, longer time horizons, more ambiguous information, less certain constraints, and deeper conflicts among aims. Lower-level work is more often solvable through clear pathways, sequencing, monitoring, and correction. At levels I-IV problems tend to self-identify through day-to-day processes. At level V there a more complex problem type can emerge from the interaction of multiple complex systems introducing some paradoxical elements. At levels VI-VII work enters the territory of paradox, where the task is not to find a final answer but to hold competing necessities in balance over time.

**Description of work by level in a standard organizational structure. Exhibit 6.2**

| Level | Description of aims of work | Timespan* |
|---|---|---|
| VII<br>Global MD | Civilisational, sectoral, constitutional, or global architecture. | 10-25 years and beyond |
| VI<br>Regional CEO | Institutional architecture; stewardship of large systems over long periods. | 5-10 years |
| V<br>CEO | Enterprise or major business-unit strategy; market positioning and institutional performance. | 2-5 years |
| IV<br>General Manager | Cross-functional integration; multiple interacting functions and trade-offs. | 1 to 3 years |
| III<br>Department Head | Functional management; building capability, systems, and repeatable performance. | 3 months to 2 years |
| II<br>Supervisor | Supervisory or technical coordination; sequencing several tasks and people. | Days to 3 months |
| I<br>Operator | Direct operational work; immediate tasks; visible output. | Immediate to days |

*Time is now being compressed by AI

**Nature of corporate work by holonic level for seven characteristics. Exhibit 6.3**

| Holonic Level | Aim | Scope | Innovation | Paradox | Information Certainty | Self-governance | Time |
|---|---|---|---|---|---|---|---|
| VII<br>Global MD | Systemic, global interdependency | Global scale | Paradigm-shifting | Extreme, pervasive | Very low | Value anchored, Integrative | 10 –25 years |
| VI<br>Regional CEO | Complex systems integration | Multi-geography, multi-industry | Transformational | Paradox | Low | Reflective under paradox | 5 –10 years |
| V<br>CEO | Strategic plans | Whole enterprise | Strategic innovation | Some paradox | Moderate to low | Strategic & Principled | 2–5 years |
| IV<br>General Manager | Functional integration | Full operations / divisions | Efficiency innovation | Data ambiguity | Mixed | Accountable, standards driven | 1-3 years |
| III<br>Department Head | Process optimisation | Sub-function control | Process innovation | Little ambiguity | Mostly certain | Task focused responsible | 3 months – 2 years |
| II<br>Supervisor | Procedural troubleshooting | Small function team | Minimal | Low | High | Rule following | Days – 3 months |
| I<br>Operator | Declarative, rule-based | Single tasks | None | None | Very high | Impulse control | Immediate – days |

## 3. Ten-step process to solve levelled problems

For complex, levelled problems a ten-step problem-solving process is required. The first three steps provide the orientation that conventional problem-solving methods often assume has already occurred - **what has caused us to intervene, at what level the problem actually resides, and whether the person attempting to solve it has the capability and authority to work at that level.** These locate both the problem and the problem solver within the wider system. The remaining seven steps then take us through the more familiar sequence of **definition, analysis, solution generation, evaluation, choice, implementation** and **learning**.

The result is an extension of traditional problem solving. The central proposition is simple: before solving a problem, locate both the problem and the problem solver within the hierarchy of the system.

**Step 1. Recognise the Condition**

Problem solving begins because something causes us to become dissatisfied with the present situation. Sometimes the **trigger is negative**:

- Performance is deteriorating,
- Something has failed,
- Risk is increasing, or
- An expected outcome is no longer being achieved.

A company whose profit has fallen from 20 to 10, for example, has an evident reason to investigate. At this point, however, declining profit is not yet a properly defined problem. It is an observed condition: something important has changed and requires attention.

The **trigger can equally be positive**. Nothing may be failing, but there may be an opportunity to move to a substantially better state:

- Profit may be 20 and management may believe that it could become 40.
- A government service may be functioning adequately but could produce much better outcomes.
- A technology may create possibilities that did not previously exist.

In these cases, the starting point is not the need to stop deterioration but the desire to create improvement. Problem solving therefore includes both correcting an adverse direction and deliberately producing a more desirable one.

The discipline at this stage is to avoid prematurely explaining the condition. “Profits are falling” is an observation. “Our sales force is ineffective” is already a hypothesis about its cause. Jumping too rapidly from the first to the second can lock the entire problem-solving process onto the wrong path. The first step is simply to establish as clearly as possible, what is happening, what direction is undesirable or what improvement is sought, and why we need to act.

**Step 2. Locate the Problem by Level**

Once the condition has been recognised, the next and more difficult question is: at what level does the problem actually live? A symptom visible at one level of a complex system may be caused by forces operating somewhere else. The level at which a problem first becomes visible is therefore not necessarily the level at which it should be understood or solved.

Consider profitability. A company whose profits have declined might initially treat the issue as a profit-and-loss problem. But the explanation might sit at several quite different levels.

- The global structure of the industry may have changed;
- Domestic competitive conditions may have deteriorated;
- The company may have lost relative competitive position;
- Its business economics may no longer work;
- One function may be performing badly; or
- The difficulty may lie within a particular operational process.

Each of these is a legitimate problem space, but each requires different evidence, different capabilities and different remedies.

This makes locating the problem by level a diagnostic act in its own right. The problem solver must deliberately move up and down the hierarchy and ask whether the apparent cause is itself the consequence of something occurring elsewhere. A sales shortfall might reflect weak selling, but it might instead reflect an inferior value proposition created years earlier by strategic choices. The practical question is therefore not merely, “What is wrong?” but “At what level is the most useful explanation for what we are observing likely to reside?” Until that question is addressed, conventional analysis risks becoming highly competent work on the wrong problem.

**Step 3. Locate Yourself**

Locating the problem is only half of the orientation. The second half is to locate the problem solver. A person may identify correctly that a problem resides at a particular level while lacking either the capability or the authority to solve it there. These are different constraints and need to be considered separately.

Capability concerns the level at which someone can genuinely think and work. Higher-level problems generally contain more variables, wider relationships, longer time horizons and greater uncertainty. A manager who is highly capable at improving a manufacturing process may not yet be equipped to redesign the relative competitive position of an entire corporation. Equally, someone capable of recognising a strategic problem may not have sufficient specialist knowledge to diagnose its technical parts. Problem solving therefore requires intellectual honesty about the level of complexity that the individual or team can competently handle.

Authority concerns permission to act. A functional manager may understand that a problem originates in corporate strategy but have no legitimate authority to change that strategy. In that case, the correct response is not to pretend that the problem exists only within the function. It may require escalation upward, collaboration across boundaries, decomposition into parts that can be acted upon locally, or the involvement of people with the appropriate mandate. Effective problem solving therefore seeks fit among three things: the level at which the problem resides, the level at which the problem solver can understand it, and the level at which the problem solver is authorised to act.

**Step 4. Define the Problem**

Only after the condition, problem level and problem solver have been located can the problem be defined with useful precision. Problem definition converts a broad concern into a statement that can actually be investigated. It should describe the gap between what is occurring and what is required, while avoiding unsupported assumptions about why the gap exists.

For example, “profits are falling” remains too broad. “Manufacturing costs are too high” may already be too narrow if manufacturing has not yet been shown to be the principal cause. A better formulation might be: “Our domestic business's profit has fallen from 20 to 10 over three years. We need to determine whether the deterioration arises principally from industry economics, relative competitive position, business economics or functional execution.” That statement gives the investigation boundaries without predetermining the answer.

A well-defined problem should therefore specify the relevant outcome, time frame, scope and level, together with enough context to guide inquiry. It should also remain open to revision. Problem definition is not a ceremonial step completed once at the beginning. As evidence accumulates, the framing may need to narrow, broaden or move to a different level. Good problem solvers redefine the problem when the investigation shows that the original framing was incomplete or wrong.

**Step 5. Analyse the Problem**

Analysis asks what is causing the defined condition. The purpose is to move from observation to explanation. This involves breaking the problem into its constituent parts, identifying patterns and relationships, gathering relevant evidence, testing hypotheses and distinguishing root causes from symptoms or correlated events.

Analysis should remain conscious of levels. Causes may operate vertically as well as horizontally. A weak operational result may arise from a poorly designed process, but that process may itself be the result of organisational incentives, strategic choices or constraints imposed by the level above. Conversely, a high-level problem may sometimes be the aggregate consequence of many local failures. Analysis therefore needs to follow causal chains far enough to identify where effective action is possible rather than stopping at the first plausible explanation.

The quality of analysis depends heavily on the information available. Separate claims from evidence. Check the source, method, reliability and counterevidence. Test interpretations against alternatives. The aim is a sufficiently accurate model of the problem to direct the solution at causes that matter.

**Step 6. Generate Possible Solutions**

Once the causes of the problem are sufficiently understood, the task changes from diagnosis to design. The question becomes: what could we do? At this stage, premature commitment is dangerous. The aim is to create a range of credible alternatives rather than allowing the first plausible solution to dominate simply because it appeared first.

Solution generation benefits from different perspectives. Different functions, disciplines and levels of the organisation may see different places to act. One option may attempt to eliminate the root cause directly. Another may reduce its consequences. A third may change the surrounding system so that the cause becomes less important. Some solutions may operate locally; others may require action at L+1 or coordination across several holons. Creativity therefore comes partly from changing the level at which the action is considered.

The possible solutions should nevertheless remain grounded in the diagnosis. Constrain brainstorming. The most productive question is not merely “What ideas do we have?” but “Given what we now understand about the causes, what different mechanisms could plausibly move the system toward the desired condition?” This keeps creativity connected to causality.

**Step 7. Evaluate the Alternatives**

Generating alternatives expands the field; evaluation narrows it. Each option must now be tested against the aims, constraints and likely consequences of the problem. This involves assessing effectiveness, feasibility, cost, risk, speed, resource requirements and the probable reactions of the wider system.

Evaluation is particularly important in complex adaptive systems because actions rarely produce only their intended effect. A solution that improves one measure may damage another. A local optimisation can move costs elsewhere. An action that succeeds in the short term may weaken adaptability over time. Alternatives should therefore be assessed not only for their direct benefit but also for trade-offs, second-order effects, reversibility and consequences at adjacent levels.

The evaluation should expose assumptions. If an option works only if customers behave in a particular way, if a regulator cooperates, or if another function supplies scarce capability, those dependencies need to be made explicit. Where uncertainty is high, run a pilot may before full commitment. The purpose of this step is to understand the relative strengths, weaknesses and risks of the available choices well enough to make an informed judgement.

**Step 8. Choose the Solution**

Choice converts analysis into commitment. At some point, the organisation has to decide which option, or combination of options, it will pursue. The best choice is not necessarily the solution with the largest theoretical upside. It is the one that most appropriately addresses the problem given the evidence, aims, constraints, capabilities, authority and acceptable risk.

Decision-making should therefore return explicitly to the original framing. Does the proposed solution act at the level where the problem was found to reside? Does the person or institution making the decision possess the necessary authority? Does implementation require coordination at L+1 or with neighbouring systems? Has the option been chosen because the evidence supports it, or because it is familiar, politically convenient or favoured by the most powerful participant?

A decision should also state what success will look like. Choosing a solution without identifying the expected results makes later learning difficult. The decision should establish the principal outcome sought, relevant operating ranges, assumptions to be monitored and circumstances that would cause the decision to be revisited. In this sense, choosing a solution begins the control-and-correction cycle that continues through implementation and review.

**Step 9. Implement**

A good decision has no value unless it can be converted into coordinated action. Implementation translates the chosen solution into responsibilities, resources, timing, milestones, operating processes and behavioural changes. This is the point at which an intellectual solution encounters the reality of the system.

Implementation frequently fails because the solution is treated as a single instruction rather than as a change to an interconnected system. People may need new capabilities. Incentives may need to change. Information may need to move differently. Existing processes may resist the action. Actions at one level may depend on permissions or resources controlled at another. Effective implementation therefore asks not simply, “What must be done?” but “What conditions must exist throughout the system for this decision to become real?”

The implementation plan should also include sensing and feedback from the beginning. Progress measures, exceptions, unexpected effects and leading indicators should be visible early enough to allow adjustment. Implementation is therefore not merely execution of a fixed answer. It is disciplined action accompanied by continuous observation.

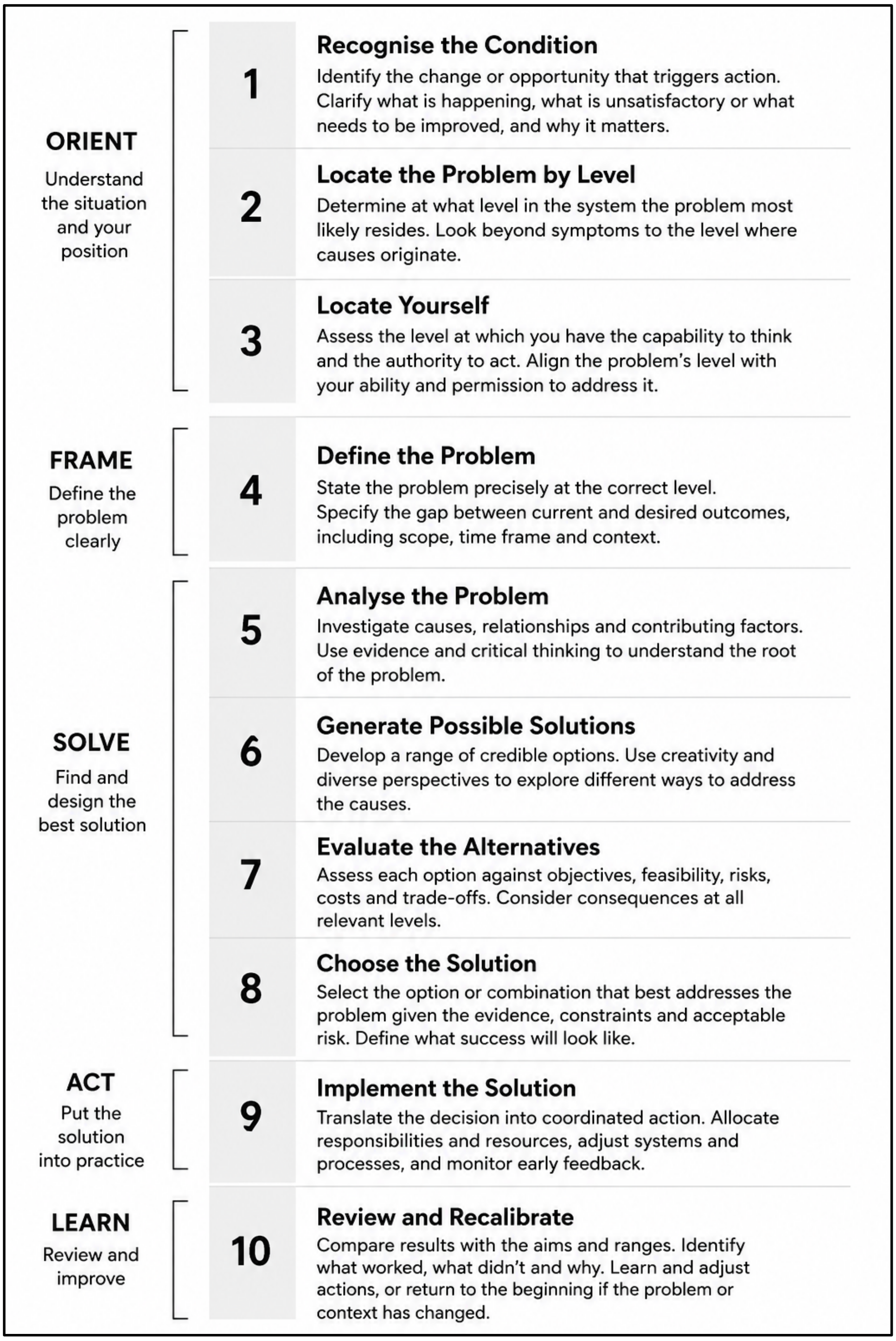
**Step 10. Review and Recalibrate**

The final step is to determine what happened, why it happened and what should now change. Results should be compared with the aims and ranges established when the solution was chosen. Did the action produce the intended effect? Did it generate unexpected consequences? Were the original assumptions correct? Has the environment changed while implementation was under way?

Review turns problem solving into learning. Failure does not automatically mean the chosen solution was foolish, just as success does not prove that every underlying assumption was correct. The result may reflect implementation quality, external events, incorrect diagnosis, inappropriate problem level, or simply uncertainty inherent in the system. The task is therefore to distinguish among these possibilities rather than reducing the review to a binary judgement of success or failure.

Correction then closes the loop. The system may need to alter its actions, modify the solution, revise its aim, improve its sensing, change its decision rules—or return all the way to the beginning and reconsider the level at which the problem was defined. In complex systems, problem solving is therefore rarely linear. Review may reveal that the problem has moved, that the problem solver must move, or that the original condition was misunderstood. The ten-step process is best understood as a recursive learning cycle: orient, frame, solve, act, learn—and then begin again at the level the new evidence requires.

**The 10-step problem solving process for complex, levelled problems. Exhibit 6.4**



## 4. Solvable problems

A solvable problem has an aim that can be described, evidence that can be gathered, and a pathway that can be tested. The answer may be difficult, uncertain or expensive, but in principle the system can determine whether the problem has been solved. A machine can be restored to operation, a production plan can be completed, a bridge can be designed to carry a specified load, and a process can be changed to meet a defined standard.

The cybernetic structure developed in the previous chapter is especially effective here. The system senses the present state, compares actual with intended performance, acts to reduce the gap and uses feedback to determine whether the correction worked. As the level rises, the problem becomes more ambiguous and the pathway less obvious, but Levels I to V remain largely within this solvable territory.

## 5. Maintenance as a levelled solvable problem

We use maintenance as a demonstration of how all processes present solvable problems which can be can be levelled. Maintenance is a particularly useful example because physical deterioration eventually reports the truth. Bearings wear, pipes corrode, concrete cracks and machines stop. The work can therefore be recognised through eleven invariant nouns. The nouns stay stable; problem solving changes by level.

**Eleven invariant nouns of maintenance. Exhibit 6.5**

| # | Invariant noun | Permanent question |
|---|---|---|
| 1 | Asset | What object, system or architecture is being maintained? |
| 2 | Aim | What condition or performance must it sustain? |
| 3 | Standard | What counts as acceptable and unacceptable condition? |
| 4 | Information | What must be known about condition and performance? |
| 5 | Monitoring | How is actual condition observed against the standard? |
| 6 | Failure | What can deteriorate, deviate or break? |
| 7 | Trigger | What condition initiates intervention? |
| 8 | Time | When must attention or intervention occur? |
| 9 | Intervention | What is done to restore, preserve or improve performance? |
| 10 | Resources | What people, skills, parts, tools, money and time are required? |
| 11 | Review | What is learned and changed after observing outcomes? |

At Level I, the system is a single item: it breaks, somebody notices, it is repaired or replaced, and the immediate question is whether it works again. At Level III, the object is an asset system: condition is measured through temperature, vibration, wear or pressure, and action starts before predicted failure. At Level IV, the object is an interdependent operating system: recurring failure patterns and consequences are analysed, risk becomes the trigger, and the action aims to eliminate the underlying cause. At Level V, continuous data and prediction allow the system to detect emerging failure signatures and intervene dynamically.

## The maintenance levels matrix. Exhibit 6.6

| Level | Asset | Aim | Standard | Information | Monitoring | Failure | Trigger | Time | Intervention | Resources | Review |
|---|---|---|---|---|---|---|---|---|---|---|---|
| VII Maintenance by design | System-of-systems / architecture | Remove need for maintenance failure | Long-term survivability and regeneration | Structural lifecycle patterns | Observe architecture and emerging conditions | Failure-generating design | Future unsustain-ability | Generational / lifecycle | Redesign before failure emerges | Science, design, governance, long-horizon capital | What should never need maintaining again? |
| VI Adaptive / Self-optimising | Network of systems | Sustain whole-system performance | Dynamic system constraints | System-wide flows, interactions and risks | Monitor system and environment | Interaction or architecture failure | Trajectory moves off course | Multiple horizons | Reconfigure / redistribute / redesign | Networked capability and redundancy | Did the whole system adapt? |
| V Predictive / Self-diagnosing | Complex operating system | Prevent disruption | Predicted performance envelope | Continuous multi-source data | Automated sensing and diagnosis | Emerging failure signature | Model predicts deviation | Dynamic / real time | Predictive intervention | Integrated data, AI and specialists | Was prediction accurate? |
| IV Reliability-based | Interdependent assets | Sustain system reliability | Reliability / availability target | Failure patterns and consequences | Analyse trends and dependencies | Recurring or systemic failure | Risk exceeds tolerance | Optimised window | Eliminate recurring cause | Cross-functional engineering resources | Did reliability improve? |
| III Condition-based | Asset system | Maintain condition | Condition limits | Temperature, vibration, wear, pressure | Measure condition | Deterioration | Limit approached | Before predicted failure | Targeted maintenance | Diagnostic skill and specific parts | Did condition recover? |
| II Time-based | Similar assets | Prevent breakdown | Service specification | Age, hours, cycles | Record elapsed use | Expected wear | Due date / interval | Fixed calendar | Inspect / service / replace | Scheduled labour and parts | Was service completed? |
| L I Breakdown | Single item | Make it work | Works / does not work | Visible problem | Notice breakdown | Broken | Stops working | After failure | Repair / replace | Person, tool, spare | Did it work? |

The concept of levelled maintenance can be seen in vehicle maintenance. The vehicle does not itself possess a maintenance level. The level lies in the problem-solving architecture applied to it. The same car can be repaired after failure, serviced by calendar, monitored by condition, managed for reliability or embedded in a predictive maintenance system. As the level rises, the object of attention expands from the failed part to the vehicle, then to interacting systems, fleets and ultimately the design architecture that generates the need for maintenance.

**Vehicle maintenance by level. Exhibit 6.7**

| **Level of Maintenance** | **What vehicle maintenance looks like at each level** |
|---|---|
| VII Maintenance-by-design | The question moves beyond maintaining existing vehicles: what failures should future vehicle and transport architectures eliminate altogether? Components, software, modularity, materials and infrastructure are redesigned so recurring classes of maintenance become unnecessary or radically reduced. |
| VI Adaptive / self-optimising | At this level the object is no longer one car but a connected transport or fleet system. Maintenance resources, charging, parts, software, vehicle utilisation and intervention timing are dynamically reconfigured so the **whole system remains available and efficient**. |
| V Predictive / self-diagnosing | Continuous sensor and operating data are analysed to **detect emerging failure signatures**. Software predicts that a battery, bearing, cooling component or drivetrain element is moving outside its expected performance envelope and schedules intervention before disruption. |
| IV Reliability-based | Repeated faults are analysed across the vehicle or fleet. If a component repeatedly fails, the question shifts from “replace it again” to “why does this failure recur?” Maintenance seeks to **eliminate the underlying cause** and improve whole-vehicle reliability. |
| III Condition-based | Actual **condition is measured**. Tyre tread, brake-pad thickness, oil condition, battery health, fault codes, temperatures and other indicators trigger maintenance before failure occurs. |
| II Time-based | The vehicle is **serviced according to kilometres, elapsed time or prescribed intervals**. Oil, filters, fluids and components are inspected or replaced according to the manufacturer's schedule whether or not deterioration is yet evident. |
| I Breakdown | The car fails: flat battery, broken belt, failed starter, puncture. The fault is noticed after failure and the **defective item is repaired or replaced**. Success means the car works again. |

## 6. When problem solving becomes paradox-holding

Solvable problems dominate Levels I–V; paradox begins to appear at the upper end of V and becomes characteristic at VI–VII. Paradoxes are two necessary goods that pull in different directions, where maximising either side eventually damages the whole. Our aim is not a single solution to a problem, rather determining what is the best balance between two states.

This is the first major boundary in the levels of problem-solving framework. Paradox will be explored in depth in chapter 6.

## 7. What this means for humans

The argument is now cumulative. Chapter 1 showed that we live inside nested wholes. Chapter 2 showed that those human systems are animated by survival brains. Chapter 3 showed that those brains act on truth and felt truth. Chapter 4 showed that any system that wants to survive needs aims, sensing, comparison, correction and learning. This chapter adds the next step: before we solve anything, we have to know what we are trying to achieve and at what level the problem sits.

That sounds obvious. It is not. We often start with the symptom we can see, the job title we recognise or the action we know how to take. The result can be excellent work on the wrong problem.

The discipline is therefore simple. Start with the aim. Recognise the condition. Locate the problem by level. Locate yourself. Do you have the capability to understand the problem? Do you have the authority to act at that level? Only then define, analyse, choose, act and learn.

The higher the level, the harder this becomes. More parts have to be held together. The time horizon gets longer. Evidence becomes less direct. More people carry different interests and felt truths. At the highest levels, even the aim may be contested.

That is where solvable problems begin to give way to paradox. Freedom and order both matter. Competition and cooperation both matter. Present needs and future responsibility both matter. The human task is no longer to eliminate one side. It is to hold the tension and keep correcting the balance as reality changes.

This gives us a practical form of problem-solving literacy. Ask: What is the aim? Where does the problem live? What is physical? What is relational? What do we know? What do we only feel to be true? What can be solved? What has to be held? What happened? What did we learn?

The aim is not to make every human a strategist. It is to help us see the level of the problem before we rush to solve it. That gives us a better chance of choosing the right work, the right people and the right form of correction.

## 8. What this means for AI

AI now enters a clearer architecture. It works inside nested systems, with human brains, human felt truths, control-and-correction loops and levelled problems. Its power depends on where it is placed and what aim it is given.

For bounded problems, AI can be extraordinarily useful. It can gather evidence, compare options, find patterns, diagnose causes, model alternatives, monitor performance and learn from results. Maintenance shows the direction clearly: from noticing failure, to sensing condition, to predicting failure, to redesigning the system that creates it.

But AI does not remove the need to locate the problem. A powerful answer at the wrong level can make the larger system worse. AI can optimise a task while damaging the workflow. It can optimise a function while damaging the organisation. It can optimise one organisation while damaging the wider society.

Nor should AI own the aim. A system can optimise only toward the aim it has been given. If the aim is too narrow, badly framed or set at the wrong level, greater intelligence simply gets us to the wrong place faster.

The boundary becomes especially important when work becomes paradoxical. AI can show consequences. It can model choices. It can surface hidden assumptions and felt truths. It can help us see who wins, who loses and what happens over time. But the legitimate settlement of freedom and order, competition and cooperation, national and global, present and future is still a human governance task.

So the role of AI is not simply to solve problems faster. It is to help us see the problem, the level, the evidence, the trade-offs and the consequences more clearly. Used that way, AI strengthens human judgement and correction. Used badly, it makes a mistaken aim more powerful.

The rule is simple: humans set and legitimate the aim; AI helps us understand, act, sense and learn; both remain inside a control-and-correction system that can see when reality has changed.