Five Strands *of* Diagnostics, Observability, and Debugging

Dmitry Vostokov, DumpAnalysis.org

A TECHNOLOGICAL AND INTELLECTUAL SYNTHESIS

Companion overview to the five linked histories

Concept and direction by Dmitry Vostokov, DumpAnalysis.org

Developed with AI assistance from GPT-5.6 Thinking (ChatGPT interface label, accessed 2 August 2026) and Claude Opus 5 Extra (interface label, accessed 2 August 2026); provider build identifiers were not exposed by either interface. Dmitry Vostokov set the scope and argument, selected the source base, directed all revisions, and reviewed the manuscript. AI assisted with drafting, comparison, and copy-editing; factual and technical claims added during this revision were checked against the cited sources and the five companion histories.

August 2026

Abstract

The five histories of software, hardware, medical, machine-learning and artificial-intelligence, and technical diagnostics describe different objects, institutions, and standards of proof. Yet they share one enduring problem: important state is hidden, observations are partial, and action must be justified before certainty is complete. Across bodies, devices, programs, learned models, and engineered assets, diagnosis turns signs into evidence, evidence into competing explanations, explanations into tests or interventions, and outcomes into revised knowledge [1–5].

This synthesis treats the five strands as a family rather than a hierarchy. Medical diagnostics begins from the person and a care question; hardware diagnostics reasons across matter, circuitry, encoded state, and serviceability; software diagnostics reconstructs execution and distributed causality; ML and AI diagnostics examines data, training, model behavior, production feedback, and agent trajectories; technical diagnostics evaluates the performance and integrity of physical structures and systems. Their differences are not defects to be normalized away. They reveal where a general diagnostic vocabulary is useful and where domain-specific evidence, ethics, and consequence must remain primary.

The central argument is that all five strands can be organized around the canonical diagnostic continuum developed in Table 5: Observe; Measure / preserve; Contextualize; Compare; Infer; Discriminate; Act; Verify and learn. Observability supplies the designed evidence surface; diagnostics discriminates among explanations; debugging is the interventionist search for where and how behaviour departs from intent; monitoring follows selected conditions over time; prognosis extends the argument into the future. No single signal or model completes this continuum, and AI strengthens it only when its claims remain grounded, auditable, reversible, and subject to accountable human–automation design. A further, explicitly provisional unification is proposed through analysis patterns: reusable operations for comparing, reconstructing, correlating, discriminating, and verifying evidence without erasing the different objects, standards of proof, and ethical duties of the five strands. This proposal is demonstrated in Appendix C, compared with established rival unifiers, and given explicit conditions under which it should be rejected. The claim is deliberately narrow: diagnostic analysis

patterns may provide a useful inquiry-level layer above formal models and domain practices because they can describe what investigators do when no single structural, probabilistic, or control-theoretic formalism covers the object. Where one of those formalisms is available, the pattern should invoke it rather than compete with it. [1–5]

How to read this synthesis

This article does not compress the five histories into one origin story. The strands developed at different times and scales: medical diagnosis preserves ancient case traditions; technical and hardware diagnosis grew through mechanics, metrology, electrical testing, reliability, and service practice; software diagnosis emerged with stored programs and production systems; ML and AI diagnosis joined statistical model criticism, symbolic explanation, learning theory, MLOps, and governance. Similar forms often appeared independently, and the same term may carry different obligations in different domains [1–5].

References to the five linked companion histories point to the detailed historical source bases on which the strand summaries rest. This synthesis also cites external primary papers, standards, official specifications, and authoritative guidance directly wherever the synthesis makes a cross-strand structural claim. The pattern-oriented references document the existing Pattern-Oriented Diagnostics, Pattern-Oriented Observability, Higher-Order Pattern Narratives, and hardware-narratology lineage used in Chapters 26–30. Cross-strand claims identify structural similarities, but they should not be read as asserting that a patient, a bridge, a processor, a distributed service, and a foundation-model agent are interchangeable objects.

Five linked histories

Each strand is developed in depth in its own companion history. The titles below are live links.

Table 1. The five linked companion histories.

STRAND	PRIMARY OBJECT	CHARACTERISTIC EVIDENCE	DISTINCTIVE EMPHASIS
Medical diagnostics	Person, body, disease process, care pathway	History, examination, specimens, images, waveforms, outcomes	Patient context, probability, ethics, communication, and clinical utility
Hardware diagnostics	Electronic and computing devices across physical and logical layers	Waveforms, test patterns, syndromes, counters, telemetry, physical failure analysis	Fault containment, first-failure capture, cross-layer matter and representation
Software diagnostics	Programs, executions, services, and organizations	Dumps, logs, metrics, traces, profiles, code, configuration	Reconstructing invisible and often distributed execution under partial evidence

STRAND	PRIMARY OBJECT	CHARACTERISTIC EVIDENCE	DISTINCTIVE EMPHASIS
ML and AI diagnostics	Data-model-pipeline-service-human systems	Residuals, splits, gradients, explanations, drift, evaluations, trajectories	Systems can fail while executing correctly; behavior is statistical, adaptive, and governed
Technical diagnostics	Physical structures, machines, plants, and engineered assets	NDE, strain, vibration, thermography, condition histories, SHM, failure analysis	Integrity, degradation physics, qualification, and consequence-based intervention

Neighboring practices and their primary questions

Table 2. A unified working vocabulary; real investigations combine several practices.

PRACTICE	PRIMARY QUESTION	CHARACTERISTIC EVIDENCE
Observation / inspection	What can be directly noticed, elicited, or measured now?	Signs, dimensions, images, waveforms, physical condition, narratives
Testing / measurement	Does the object, specimen, model, or system satisfy a stated expectation under defined conditions?	Stimuli, reference standards, test vectors, assays, limits, calibration records
Monitoring	How is selected condition changing, and when does it require attention?	Trends, telemetry, repeated specimens, counters, alarms, trajectories
Diagnostics	Which state or mechanism best explains the evidence?	Correlated observations, competing hypotheses, residuals, causal tests, context
Debugging	Where and how does behavior depart from intent, and what changes under intervention?	Breakpoints, substitutions, controlled perturbations, replays, targeted tests
Observability	What hidden behavior can be inferred from designed outputs and preserved context?	Logs, traces, metrics, images, signals, provenance, identifiers, models
Prognosis	How may condition evolve, and when may intervention become necessary?	Degradation models, outcomes, uncertainty bounds, remaining-life estimates
Forensics / diagnostic review	What happened, in what sequence, and what was missed or misunderstood?	Preserved artifacts, timelines, chain of custody, autopsy or teardown, postmortems

SEVEN RECURRING CROSS-STRAND TRANSITIONS

- Sensory signs and local observations → quantified variables, structured records, and calibrated instruments
- One-time tests and snapshots → longitudinal histories, trajectories, and condition-aware baselines
- External examination → embedded, digital, remote, and continuously queryable observation
- Detection of anomaly → localization, mechanism, consequence assessment, and prognosis
- Individual cases and components → systems, populations, fleets, infrastructures, and organizations
- Unaided expert interpretation → AI-assisted ranking, explanation, and action under explicit governance

- Tool-specific artifacts → interoperable evidence with stable identity, time, provenance, and semantics (Ch. 31)

Contents

Seven parts · 33 chapters · three appendices · one synthesis figure · linked references in first-citation order

Table 3. Article contents and opening pages.

SECTION	PAGE
PART I · FRAMING THE FIVE STRANDS	**7**
1. Why five histories?	7
2. Five diagnostic objects	7
3. Shared vocabulary and limits of analogy	8
PART II · THE STRANDS IN THEIR OWN TERMS	**9**
4. Medical diagnostics	9
5. Hardware diagnostics	10
6. Software diagnostics	10
7. ML and AI diagnostics	11
8. Technical diagnostics	12
PART III · THE SHARED HISTORICAL ARC	**13**
9. From signs to quantified evidence	13
10. From snapshots to trajectories	13
11. From external examination to embedded and continuously queryable observation	14
12. From detection to localization, mechanism, consequence, and prognosis	14
13. From local artifacts to systems and populations	14
14. From human craft to AI-assisted diagnosis	15
PART IV · A UNIFIED EVIDENCE ARCHITECTURE	**16**
15. The diagnostic continuum	16
16. Identity, provenance, and time	17
17. Baselines, thresholds, and reference standards	17
18. Probability, loss, and the value of a test	18

SECTION	PAGE
19. Causal isolation, intervention, and verification	18
20. Uncertainty, consequence, and action	19
PART V · CONVERGENCE WITHOUT COLLAPSE	**20**
21. What the strands borrow from one another	20
22. Boundary zones and overlapping objects	20
23. Irreducible differences	21
PART VI · TOWARD A GENERAL DIAGNOSTIC DISCIPLINE	**22**
24. Shared principles	22
25. A cross-strand evidence taxonomy	23
26. Analysis patterns as a candidate unifying layer	24
27. Comparison with other unifiers and what would refute the proposal	27
28. AI as instrument, subject, and governor	28
29. Reflexive diagnostics: diagnosing observability, diagnostics, and debugging	29
30. Pattern narratives and narratological unification	30
PART VII · FUTURE DIRECTIONS AND CONCLUSION	**32**
31. Interoperable evidence ecosystems	32
32. Governed diagnostic agents and digital twins	32
33. Conclusion	33
APPENDIX A · COMPARATIVE MATRIX	**35**
APPENDIX B · SHARED GLOSSARY	**36**
APPENDIX C · WORKED CROSS-STRAND PATTERN CASES	**40**
REFERENCES	**44**
FIGURE 1 · FIVE-STRAND SYNTHESIS	**46**

PART I

Framing the five strands

1. Why five histories?

A single history of diagnostics would either become so general that it lost contact with evidence or so narrow that it mistook one domain for the whole field. The five-strand approach avoids both errors. It follows diagnostics wherever hidden state must be inferred, but lets each domain define what counts as a subject, a failure, a permissible intervention, and an adequate explanation [1–5].

The strands also expose different sources of diagnostic authority. Medicine combines the patient's account, clinical judgment, tests, population evidence, and outcomes. Hardware combines design intent, electrical or logical state, telemetry, service records, and physical failure analysis. Software combines formal artifacts with contingent execution and organizational change. ML and AI add data-generating processes, learned representations, evaluation design, feedback loops, and agent side effects. Technical diagnostics combines mechanics, materials, NDE, structural integrity, maintenance, and lifecycle assurance.

Seen together, the histories reveal that diagnostic progress is rarely the invention of one sensor. It is the construction of an evidence system around the sensor: identity, calibration, baseline, coverage, interpretation, escalation, and feedback. A new instrument widens the observable surface while creating new opportunities for artifact, drift, misuse, and false confidence. The same dialectic appears in the stethoscope, oscilloscope, memory dump, CT scanner, gradient dashboard, vibration monitor, and agent trace.

2. Five diagnostic objects

Table 4. The object of diagnosis changes the meaning of evidence and intervention.

STRAND	DIAGNOSTIC OBJECT	HIDDEN STATE OR FAILURE OF INTEREST
Medical	A person within a biological, social, and care context	Physiology, pathology, disease extent, risk, response, and missed or delayed care
Hardware	A physical device or platform whose state is also encoded logically	Material defect, electrical margin loss, erroneous state, containment failure, and degradation
Software	A formal artifact executing in a changing technical and organizational environment	Incorrect state, unintended control flow, resource interaction, distributed causal sequence, and operational failure
ML and AI	A data-model-pipeline-service-human system that learns, predicts, generates, or acts	Miscalibration, leakage, shortcut learning, drift, unfairness, hallucination, unsafe trajectory, and governance failure

STRAND	DIAGNOSTIC OBJECT	HIDDEN STATE OR FAILURE OF INTEREST
Technical	A physical structure, machine, plant, process, or cultural object	Damage, integrity loss, degradation mechanism, functional departure, and future condition

The objects overlap but are not identical. Hardware diagnostics focuses on electronic and computing devices, including the interfaces through which physical faults become logical errors. Technical diagnostics has a broader engineering scope of structures, machines, process equipment, infrastructure, and physical assets; the technical history deliberately leaves computer-device diagnosis to the hardware strand except where computing mediates physical measurement [2, 5].

Software and ML systems overlap operationally, yet the ML and AI strand adds a distinctive failure class: the program may execute as written while the learned behavior is wrong, brittle, miscalibrated, unfair, or unsafe. Conversely, software faults in serving, orchestration, retrieval, or telemetry may present as an apparently model-level failure. Diagnosis must therefore widen or narrow the object without losing which claim belongs to which layer [1, 4].

3. Shared vocabulary and limits of analogy

The word diagnosis migrated among medicine, engineering, and computing because it names a recognizable reasoning form: observe a symptom, collect evidence, compare alternatives, test a hypothesis, intervene, and reassess. The analogy is productive when it carries methodological discipline. It is misleading when it implies that every domain has the same ontology, the same reference standard, or the same ethical relation to the object being examined.

Debugging is similarly broad. In software and embedded systems it often means direct control of execution. In hardware and technical practice it can mean varying load, substituting a module, sectionalizing a system, or reproducing a physical symptom. In medicine, the companion history uses debugging only as a carefully limited metaphor for iterative hypothesis testing and correction of diagnostic workflows, not for treating a patient as a machine [3].

Observability names the designed relation between hidden state and obtainable evidence. In control theory it has a formal model-based meaning. In contemporary software it emphasizes inferability from production outputs. In ML and AI it includes data, training, model internals, outputs, traces, feedback, and downstream outcomes. In physical and medical systems, the same framing is useful only when it remains attached to sensing physics, specimen paths, patient context, and measurement limitations [1, 3–5]. Control-theoretic observability is a formal property of a specified dynamic model and output relation; it is therefore a candidate unifier only where a defensible state-space model exists. [6]

Diagnosis commonly uses abductive reasoning: observations motivate one or more explanatory hypotheses, which are compared and tested rather than treated as deductions. “Best explanation” remains provisional and constrained by counter-evidence, consequence, and revision.

PART II

The strands in their own terms

4. Medical diagnostics

Medical diagnosis begins with an asymmetry between the whole person and the partial evidence available to care. The strand runs from case narratives, pulse, urine inspection, anatomy, and bedside examination through laboratory medicine, imaging, molecular testing, continuous monitoring, population surveillance, and AI-assisted diagnosis. Its central unit is not a test result but an explanatory and communicative process that integrates history, examination, prior probability, time course, alternatives, and the consequences of action [3].

Medical observability widened by scale: organs became visible through anatomy and imaging; cells and organisms through microscopy and culture; concentrations through chemistry; electrical and acoustic processes through waveforms; sequences through genomics; trajectories through monitoring; and population patterns through epidemiology. Each expansion created mediation. A specimen can be contaminated or detached from the clinical question; an image depends on acquisition and reconstruction; a reference interval can misrepresent an individual or population; an algorithm can be accurate in one workflow and harmful in another.

The strand’s distinctive discipline is the separation of analytical validity, clinical validity, and clinical utility. Measuring a target correctly is not the same as establishing its relation to disease, and both are different from improving a patient-relevant decision or outcome. Medical diagnosis also makes uncertainty, consent, equity, communication, and follow-up intrinsic to technical quality. The process is incomplete until results are understood, acted upon, and reassessed. Diagnostic-test evaluation also separates sensitivity and specificity from predictive value: the same test characteristics can lead to different post-test probabilities when prevalence or pre-test probability changes. A positive result is therefore an update to a prior state of knowledge, not an isolated verdict. [7, 8]

Concrete instance. A biomarker assay may be analytically stable and associated with disease, yet still fail to improve care if its threshold produces unnecessary procedures or if no effective follow-up is available. Conversely, a modestly informative test can be useful when it changes an urgent decision. The medical strand therefore makes context of use, benefit–risk, communication, and follow-up part of diagnostic validity rather than downstream administration.

5. Hardware diagnostics

Hardware diagnostics follows the conversion of hidden physical condition into signals, checking structures, and service evidence. It begins with sensory inspection, indicator diagrams, electrical bridges, meters, oscilloscopes, test points, and maintenance records; continues through checking circuits, error-correcting codes, redundancy, automatic test equipment, scan design, built-in self-test, firmware error records, and out-of-band management; and reaches fleet telemetry, physical failure analysis, digital twins, and AI-assisted fault isolation [2].

Its characteristic chain is fault → error → failure. A crack, radiation upset, timing margin loss, design defect, or corrupted interface may create an incorrect internal state, which may or may not produce externally visible service failure. Detection, correction, containment, diagnosis, recovery, and prognosis are separate claims. An ECC mechanism can restore a bit without establishing why it flipped; a machine check can identify a reporting bank without proving the physical cause; a replaced field-replaceable unit can restore service while leaving the mechanism unknown. [9]

Hardware diagnosis is cross-layer because matter and representation remain coupled. Electrical waveforms, logical transactions, firmware decoders, operating-system records, BMC telemetry, topology, and laboratory microscopy may each show a different point in the same causal chain. Its strongest practices preserve first-failure data, make telemetry loss visible, distinguish corrected events from health, and close the loop from field evidence to design, manufacturing, screening, and service.

Concrete instance. A rising count of correctable memory errors is evidence of detected and repaired internal error, not proof of a particular physical fault. Diagnosis may require the code syndrome, physical address mapping, DIMM identity, temperature and workload history, firmware decoding, replacement outcome, and—where recurrence matters—returned-part analysis. The operation crosses representation and matter while keeping detection, correction, isolation, and physical explanation distinct.

6. Software diagnostics

Software diagnostics is the history of making invisible execution arguable. Stored-program checking routines and selective output made mistakes inspectable; memory dumps made failure portable; symbolic and source-level debuggers made investigation conversational; static and dynamic analysis shortened the distance from defect to detection; logs, tracing, metrics, and profiles preserved behavior across production systems; distributed tracing carried context across boundaries; and SRE and postmortems turned diagnosis into an organizational practice [1].

The strand is distinguished by the coexistence of formal intent and contingent execution. Source code may be exact while the running system depends on configuration, timing, concurrency, hardware, network behavior, data, users, and organizational change. A crash dump is structurally deep but temporally narrow; a trace is sequential but selective; a metric is broad but aggregated; a log is authored; a profile samples resource attribution. Reliable diagnosis combines representations rather than seeking one universal artifact.

Software observability therefore concerns inferability, not data volume. Rich telemetry without stable identity, context propagation, semantics, collection health, and accessible raw evidence can remain diagnostically weak. The strand also makes intervention hazardous: attaching a debugger, increasing logging, restarting a process, rolling back a deployment, or issuing a production command can change the failure or create the next incident. Safe diagnosis separates restoration from explanation and preserves the evidence path through both. [10, 11]

Concrete instance. A distributed trace with a missing downstream span may indicate that the service was never called, that propagation was broken, that sampling discarded part of the path, or that the collector lost data. The trace is diagnostically useful only when propagation format, sampling policy, collector health, and deployment identity are available. This is why observation-path health belongs inside software diagnosis rather than beside it.

7. ML and AI diagnostics

ML and AI diagnostics extends statistical model criticism into systems that learn, adapt, generate, and act. Its lineages include residual analysis, statistical quality control, cybernetics, symbolic reasoning traces, holdout testing, cross-validation, benchmarks, calibration, monitoring of gradients and activations during training, interpretability, robustness, fairness, data and pipeline validation, drift monitoring, foundation-model evaluation, retrieval diagnostics, and agent trajectories [4]. [12, 13]

A defining problem is that execution success is not behavioral success. A model can run without exception while exploiting leakage, learning a shortcut, becoming overconfident, failing a subgroup, drifting after deployment, hallucinating a source, or completing a task through an unsafe sequence of tool calls. The diagnostic object expands from model parameters to data collection, training, serving, interfaces, users, feedback, and institutional controls. A benchmark score is therefore one observation, not a certificate of deployment fitness.

This strand also requires diagnostics of the diagnostic instruments themselves. Saliency maps, attribution methods, drift detectors, automated judges, sparse features—features learned from representations in which relatively few features are active for a given input—and root-cause systems are models or approximations with their own baselines and failure modes. Sanity checks, counterfactual interventions, disaggregated evaluation, human review, and adversarial testing are not optional decorations. Agentic systems further make trajectories and side effects first-class evidence: what the system was instructed to do, which tools it selected, what it observed, how memory changed, and which external state it altered.

Concrete instance. Suppose a classifier's accuracy remains stable while confidence becomes systematically too high after a population shift. A holdout comparison may detect distributional change; a calibration curve may show that reported probabilities no longer match observed frequencies; subgroup analysis may reveal that the degradation is concentrated. The model can be executing correctly in software terms while its behavioural and decision-theoretic contract has failed.

8. Technical diagnostics

Technical diagnostics concerns the performance and integrity of physical structures and systems: machines, pressure equipment, pipelines, plants, aircraft, railways, bridges, wind turbines, manufacturing processes, and cultural heritage. It grew through craft inspection, proof testing, mechanics, metrology, failure physics, nondestructive evaluation, condition monitoring, failure analysis, structural-health monitoring, prognostics, and digital twins [5]. [14]

Its evidential sequence often runs from indication to feature, anomaly, damage or fault state, mechanism, consequence, and decision. The distinctions matter. A bright thermal region is not automatically delamination; an ultrasonic indication is not automatically a rejectable defect; a vibration peak may reflect unbalance, resonance, looseness, or operating state. Qualification, probability of detection, access, calibration, geometry, and environmental normalization define what a method could have revealed. [15]

Technical diagnosis is particularly explicit about structural consequence. The same flaw size can have different meaning under different stress fields, materials, orientations, and load histories. Detection must therefore connect to fracture, fatigue, corrosion, wear, creep, fitness-for-service, or risk models. Physical intervention can also erase evidence or worsen damage, so preservation, least-invasive examination, multimodal confirmation, and auditable uncertainty are part of the engineering duty. [16]

Concrete instance. An ultrasonic indication does not become a crack-size claim merely because it is visible. Procedure qualification, reference standards, access, orientation, material, and probability of detection bound what can be inferred. A subsequent fitness-for-service assessment then combines the qualified flaw estimate with stress, geometry, material properties, environment, and projected operating conditions to support a run, repair, replace, or reinspection decision.

PART III

The shared historical arc

9. From signs to quantified evidence

All five strands begin with signs that are meaningful only against experience: a pulse quality, bearing sound, compiler failure, residual pattern, stain, vibration, temperature, or user report. Quantification does not replace skilled observation; it makes selected distinctions comparable across observers, time, and place. The thermometer, pressure indicator, strain gauge, oscilloscope, control chart, profiler, and benchmark each convert a question into a measurement architecture [1–5].

Measurement also relocates uncertainty. A number appears objective, but its meaning depends on calibration, units, sampling, preparation, reconstruction, filtering, and reference. The medical blood-pressure value depends on cuff and posture; the hardware waveform on probe and bandwidth; the software latency on clock and sampling; the ML score on split and metric; the technical indication on geometry and probability of detection. The instrument makes one hidden variable visible while making its own operation part of the evidence.

The historical advance is therefore not a march from subjective to objective. It is a move from tacit comparison toward explicit, inspectable mediation. Mature practice records the observation path, states what the method can and cannot detect, and distinguishes the measured result from the interpretation or disposition attached to it.

10. From snapshots to trajectories

Snapshots answer what state was present at a moment: a bedside examination, radiograph, memory dump, machine check, CT volume, or NDE scan. Histories answer how the state arose and whether it is changing: symptom course, vital-sign trend, event log, distributed trace, corrected-error rate, training curve, drift series, crack-growth record, or structural mode history.

The strands repeatedly discovered that present state can be ambiguous without sequence. A corrupted heap does not identify the earlier write; a current biomarker does not reveal the prior trajectory; an isolated vibration amplitude does not separate load from degradation; a final agent answer does not show unsafe tool use; a component temperature does not show whether a threshold crossing followed workload or cooling failure. Time turns evidence from description toward mechanism.

Longitudinal evidence creates its own hazards. Sensors and definitions drift, software and models change version, maintenance resets baselines, patient treatment alters physiology, and replaced components break identity. A trajectory is valid only when the points are comparable. The historical move from snapshot to history therefore required clocks, identifiers, configuration, correction records, and explicit missingness.

11. From external examination to embedded and continuously queryable observation

A recurrent historical transition moved observation from an external encounter into the object, workflow, or operating environment itself. Bedside and laboratory measurements became continuous patient monitoring; external probes and test equipment became BIST, machine-check records, and management controllers; inserted print statements became runtime tracing, metrics, profiles, and context propagation; periodic model evaluation became production drift and trajectory monitoring; scheduled technical inspection became structural-health monitoring and digital twins. The gain was temporal coverage and remote access, not automatic explanation.

Embedded observation also changes the failure surface. The sensor, firmware, instrumentation library, context carrier, sampler, collector, evaluator, or twin can fail, drift, or be misconfigured. DICOM and ASTM E2339 DICONDE show one concrete infrastructure transfer: medical imaging's identity-and-metadata envelope was adapted for nondestructive evaluation while retaining domain-specific information objects. Distributed tracing similarly depends on standardized context propagation and maintained semantic contracts. [10, 11, 17, 18]

12. From detection to localization, mechanism, consequence, and prognosis

Across the five strands, an anomaly detector answers only the first question. A diagnostic system may then localize the affected region, distinguish candidate mechanisms, estimate consequence, and project future condition. These are separate claims with different evidence requirements. A medical screen can detect elevated risk without localizing pathology; a machine check can localize a reporting bank without identifying the damaged component; a software alert can identify a failing service without locating the corrupting event; an ML monitor can detect drift without establishing its source; an NDE indication can locate a reflector without determining crack mechanism or remaining life. [9, 14, 15]

The escalation ladder is therefore: detection → localization → mechanism → consequence → prognosis. Investigations may stop at different levels when the action is already justified, but the report should name the level actually reached. A containment action can be valid before mechanism is known; a prognostic estimate requires assumptions about future load, treatment, environment, or policy that a diagnostic label does not.

13. From local artifacts to systems and populations

Early diagnosis often centered on one patient, component, program, or structure. Modern practice routinely spans care pathways, fleets, distributed services, model pipelines, infrastructures, and populations. This expansion reveals rare patterns and systemic causes, but it also changes the unit of explanation. A local anomaly may be caused by a remote dependency, organizational policy, environmental cohort, or feedback loop.

Population evidence is powerful because recurrence creates contrast. Crash buckets reveal version-specific failures; disk and memory fleets reveal concentrated error behavior; epidemiology reveals clusters and risk factors; production ML reveals subgroup regressions and drift; structural fleets reveal common details or environments. Yet aggregation can hide mechanism. A cohort association prioritizes investigation; it does not prove why a particular case failed.

The shared requirement is a return path from aggregate to artifact and from artifact to mechanism. Fleet statistics should lead to a representative dump, returned device, specimen, trace, inspection, or controlled test. Without that drill-down, a health score or anomaly cluster ranks victims but leaves the causal process untouched.

14. From human craft to AI-assisted diagnosis

Human expertise remains the historical carrier of baselines, pattern recognition, and judgment. Automation first encoded stable checks: limits, parity, rules, assertions, control charts, templates, and test vectors. Statistical methods then ranked anomalies and associations. Contemporary AI can retrieve cases, summarize heterogeneous evidence, propose hypotheses, choose queries, explain models, and operate diagnostic tools [1–5].

The progression is cumulative rather than substitutive. AI does not remove the need for calibrated sensors, correct symbols, representative data, a clinical question, a valid inspection procedure, or a trustworthy telemetry path. It inherits every upstream omission and can add persuasive unsupported inference. A fluent root-cause narrative, confident medical suggestion, or plausible mechanism label is not evidence unless it links back to observations and can survive counter-testing.

The shared design direction is bounded assistance. Diagnostic AI should expose source artifacts, distinguish observation from inference, retain alternatives, express uncertainty, seek disconfirming evidence, preserve reproducible actions, and match authority to confidence and reversibility. In high-consequence settings, human responsibility is not a ceremonial final click but the continuing ability to challenge, delay, override, and learn.

PART IV

A unified evidence architecture

15. The diagnostic continuum

Table 5. A general diagnostic continuum that can be instantiated differently in each strand.

STAGE	CORE QUESTION	TYPICAL OUTPUT
Observe	What sign, state, event, or account is available?	Narrative, image, waveform, event, sample, counter, trace, physical indication
Measure / preserve	How was it produced, and can it be retained faithfully?	Calibrated value, raw artifact, specimen, dump, immutable record, acquisition metadata
Contextualize	Which identity, configuration, environment, operating state, and time apply?	Provenance bundle, topology, patient or asset history, version, units, status
Compare	What qualified baseline, reference, model, peer, or expectation is relevant?	Residual, delta, threshold crossing, anomaly, discordance, cohort difference
Infer	Which states or mechanisms could explain the observations?	Ranked hypotheses, differential diagnosis, fault set, causal graph, uncertainty
Discriminate	What test or intervention would separate the alternatives?	Repeat or orthogonal test, perturbation, substitution, ablation, replay, controlled load
Act	What response is justified by evidence, consequence, and authority?	Treatment, repair, rollback, containment, inspection interval, abstention, escalation
Verify and learn	Did the action change the predicted outcome, and what should be retained?	Outcome evidence, regression test, postmortem, updated model, standard, or training

Table 5 is canonical for this synthesis. The same eight stages—Observe; Measure / preserve; Contextualize; Compare; Infer; Discriminate; Act; Verify and learn—are used in the abstract, worked cases, and conclusion. A real investigation may iterate, skip forward under urgency, or return to an earlier stage, but the names of the stages do not change.

The continuum is not always linear. Urgent action may precede complete explanation; a software rollback may restore service while deeper analysis continues; a medical treatment response may itself become diagnostic evidence; a hardware recovery may erase state; an ML system may require immediate suspension because the consequence is unacceptable. The value of the continuum is to label which step has been reached and which claim remains provisional.

Observability is strongest at the first four stages: it makes evidence and context obtainable. Diagnostics dominates inference and discrimination. Debugging is the controlled interaction that narrows where and how behavior departs from intent. Monitoring supplies repeated observation. Prognosis projects the state forward. Governance determines when evidence is sufficient for action and who may act.

16. Identity, provenance, and time

Every strand eventually discovers that evidence without identity is nearly unusable. A medical result needs the right patient, specimen, site, method, and status. A hardware syndrome needs physical mapping, slot, firmware, and replacement history. A software stack needs matching binaries and symbols. An ML output needs data, model, prompt, retrieval, policy, and evaluator versions. A technical indication needs asset, location, orientation, procedure, calibration, and operating condition.

Provenance is the path by which an observation became a claim. It includes acquisition, transformation, filtering, reconstruction, annotation, aggregation, and interpretation. Digital systems often make this path longer while hiding it behind a clean display. A reconstructed CT image, derived health score, normalized vibration feature, generated explanation, or symbolicated stack should remain linked to the source and processing conditions beneath it.

Time is plural. Wall-clock time relates evidence to people and external events; monotonic duration supports performance; causal order links messages or interventions; biological or degradation time may be represented by cycles, exposure, age, or treatment stage. A trustworthy synthesis preserves the notion of time relevant to the mechanism rather than forcing every event into one deceptively precise chronology.

17. Baselines, thresholds, and reference standards

A deviation exists only relative to a reference. The reference may be an individual’s prior state, a healthy peer population, a design model, a known-good unit, a validation set, a control chart, a calibration block, a service-level objective, or an acceptance code. The choice determines what becomes visible and what remains normal by definition.

The histories repeatedly warn against universal thresholds. Normal physiology varies by person and context; hardware error rates depend on topology, workload, and age; software latency depends on traffic and service objectives; ML calibration depends on population and time; structural response depends on load and environment. Thresholds compress a decision and therefore carry the costs of misses, false calls, delay, and intervention.

Qualified baselines are active artifacts. They must be versioned, maintained, and challenged after changes in instrument, population, material, configuration, workflow, or environment. A baseline that cannot change becomes an anchor to the past; a baseline that changes automatically without audit can normalize the failure.

18. Probability, loss, and the value of a test

A diagnostic observation changes a decision only through a prior state of knowledge and a consequence model. In medical testing, pre-test odds multiplied by a likelihood ratio yield post-test odds; sensitivity and specificity describe performance conditional on the target state, while predictive values depend on prevalence and selection. The same separation matters elsewhere: a rare hardware syndrome has different implications in a vulnerable lot than in a general fleet, and an anomaly score learned in one operating regime may have little meaning after deployment shift. [7, 8]

Physical inspection has an analogous but not identical apparatus. Probability-of-detection studies estimate how inspection-system response varies with flaw size and conditions, including uncertainty; fitness-for-service assessment then combines the inspection result with structural consequence and remaining-life assumptions. In ML, calibration asks whether predicted probabilities match observed outcome frequencies, not merely whether classifications are accurate. [13, 15, 16, 19]

Action thresholds should minimize expected loss rather than maximize one accuracy statistic. A rule-out threshold may tolerate false positives to reduce dangerous misses; a destructive teardown may require stronger evidence than a reversible software rollback; an autonomous agent should abstain when expected harm from an unsupported action exceeds the cost of escalation. Costs include injury, mission loss, delay, false calls, privacy, operator time, compute, storage, and missed opportunity.

Probability and decision theory are powerful cross-strand unifiers, but they do not solve every diagnostic problem. They require events, outcomes, and losses that can be represented credibly; they do not by themselves reconstruct an unknown causal sequence, establish specimen or component identity, guarantee ethical consent, define authority, or reveal that the observation path has been corrupted. The analysis-pattern layer proposed later is therefore positioned above and around probabilistic reasoning, not as its replacement.

19. Causal isolation, intervention, and verification

Correlation narrows attention; intervention distinguishes mechanisms. Medicine repeats a test, changes treatment, obtains an orthogonal modality, or follows outcome. Hardware varies voltage, load, temperature, route, or component. Software replays an execution, toggles a feature, minimizes an input, or compares a healthy cohort. ML ablates a feature, retrains without data, perturbs an input, changes a prompt, or evaluates a counterfactual. Technical diagnostics applies controlled load, targeted NDE, excavation, teardown, or laboratory examination.

An intervention is only informative when its predicted consequences are stated in advance. Otherwise any change can be narrated after the fact. Good diagnostic design records the hypothesis, expected observation, safety boundary, rollback, and evidence to preserve before action. It also considers observer effects: the debugger changes scheduling, the probe loads a circuit, the biopsy changes the specimen, the load test may extend damage, and the remediation changes future data.

Verification closes the distinction between a plausible explanation and an effective correction. A repaired part, patched service, retrained model, treatment, or maintenance action should change the predicted symptom or outcome without creating unacceptable new harm. Regression tests, follow-up, repeated monitoring, outcome review, and return-part analysis turn a local fix into durable knowledge.

20. Uncertainty, consequence, and action

Diagnostic uncertainty is not a failure to be hidden. It records limits in measurement, coverage, model, reference standard, and future conditions. The same confidence can justify different actions because consequences differ. A low-probability catastrophic crack, a possible missed cancer, a rare silent computation error, a suspected security compromise, and an uncertain model bias require different thresholds and safeguards.

The five strands express consequence through different institutions: clinical benefit and harm, safety and availability, service impact and user trust, fairness and accountability, structural integrity and risk. A general diagnostic system must therefore keep the consequence model beside the evidence. An anomaly score without a response path is decorative; a confident label without authority boundaries is dangerous.

Action can include abstention. A model may decline to predict; a diagnostic instrument may report insufficient quality; a clinician may defer a categorical label; an asset may require restricted operation; an automated agent may escalate rather than remediate. Abstention is not ignorance when it is explicit, routed, and connected to the next evidence-gathering step.

PART V

Convergence without collapse

21. What the strands borrow from one another

The strands have always exchanged concepts and instruments. Computing borrowed diagnostic language from medicine and engineering. Control-theoretic observability entered software and ML vocabulary. Reliability methods such as FMEA, fault trees, redundancy, and fault containment crossed hardware, technical systems, aerospace, safety-critical software, and AI assurance. Statistical quality control connects laboratory practice, manufacturing, drift monitoring, and production operations.

Imaging shows a more literal exchange. DICOM makes medical images inseparable from patient, study, series, acquisition, and representation metadata. ASTM E2339-21 DICONDE harmonizes nondestructive-evaluation information objects with DICOM so that industrial images and technique parameters remain portable and interpretable. This is a documented infrastructure transfer rather than an asserted resemblance: the shared envelope moves, while the diagnostic meaning remains domain-specific. [17, 18]

Software observability now supplies the operational substrate for ML and AI systems: distributed traces, metrics, logs, profiles, context propagation, semantic conventions, incident response, and rollback. ML and AI in turn assist every strand by classifying images, detecting anomalies, ranking faults, summarizing records, predicting degradation, and proposing tests. The synthesis is recursive: diagnostics uses AI, AI becomes an object of diagnosis, and AI infrastructure requires hardware and software diagnosis. [10, 11]

22. Boundary zones and overlapping objects

Table 6. Boundary zones where no single strand is sufficient.

BOUNDARY	WHY IT IS SHARED	EVIDENCE THAT MUST BE JOINED
Medical device failure	Patient outcome depends on hardware, software, calibration, workflow, and interpretation	Device logs, waveforms, configuration, clinical record, maintenance, human factors, outcome
AI infrastructure incident	Model behavior and job progress depend on accelerators, networks, storage, orchestration, and data	GPU/HBM telemetry, fabric traces, system logs, checkpoints, loss curves, topology, workload phase
Autonomous vehicle or robot	Physical sensing, embedded hardware, control software, learned perception, and environment interact	Sensor data, hardware errors, software traces, model outputs, control actions, scene and maintenance history
Digital twin decision	A maintained model mediates between a physical asset and an intervention	Asset identity, sensor quality, model version, calibration, uncertainty, predicted and observed consequence

BOUNDARY	WHY IT IS SHARED	EVIDENCE THAT MUST BE JOINED
Hospital or industrial cyber-physical system	Safety and service depend on physical process, devices, networks, software, and organization	Process variables, alarms, device and network evidence, procedures, changes, staffing, response timeline

Boundary diagnosis requires layered claims. A software exception may be the first visible consequence of hardware corruption; an ML alert may be caused by training-serving skew; a clinical error may begin with specimen identity or interface design; a technical anomaly may be a drifting sensor; a hardware replacement may hide a software or environmental trigger. Expand the investigation while further expansion can change the justified action, the responsible authority, or the prevention. Stop when additional causal levels would not change any of those three, while recording that deeper causes may still exist. Different strands legitimately stop at different levels because their intervention rights and consequences differ.

Shared boundaries also make ownership visible. When evidence crosses teams, institutions, or vendors, missing identifiers and incompatible semantics become causal obstacles. Interoperability is therefore not merely convenience; it is a condition for reconstructing what happened across the actual system boundary.

23. Irreducible differences

Medicine cannot be reduced to engineering maintenance because the patient is a person with agency, values, rights, and lived experience. Clinical action changes not only a system state but a life, and the explanatory obligation includes communication and consent. Technical and hardware objects can often be disassembled or sacrificed for analysis; people cannot. Even postmortem evidence is bounded by ethics, law, and incomplete reference standards.

Software has unusual reproducibility and formal structure. A program can sometimes be cloned, instrumented, minimized, replayed, and exhaustively reasoned about within a model. Yet production behavior can be more ephemeral than physical evidence because containers disappear, state is overwritten, and distributed causal order is partial. Hardware and technical systems retain material traces but may be inaccessible, altered by failure, or destroyed by examination.

ML and AI systems are distinctive because the behavior is learned, population-dependent, and often probabilistic. The same code and model can produce different outputs under sampling, changing retrieval, feedback, or agent environment. A “fix” can redistribute errors rather than remove them. Evaluation therefore requires multidimensional and continuing evidence rather than one conformance test.

Technical diagnostics is governed by physical consequence and qualification: detection capability, flaw geometry, degradation rate, and structural load determine action. Hardware adds encoded state and checking, while software can change behaviour without repairing matter. These differences support dialogue, not one universal method.

PART VI

Toward a general diagnostic discipline

24. Shared principles

Start from the decision and the object. State whether the task is screening, detection, localization, diagnosis, prognosis, repair, treatment, containment, or learning. A technically excellent observation can answer the wrong question.

Capture before interpretation. Evidence is perishable: physiology changes, buffers wrap, machines reset, services redeploy, models update, surfaces oxidize, and scenes are disturbed.

Preserve identity, provenance, and time. Every artifact should remain linked to the exact patient, asset, component, build, model, specimen, configuration, method, and relevant chronology.

Separate observation, inference, and action. A signal or score is an observation; a mechanism is an inference; a treatment, repair, rollback, or restriction is an action. Each requires a different standard of support.

Combine independent evidence. Agreement across modalities with different failure modes is stronger than repetition of one biased path. Disagreement should be preserved because it may reveal the observer or model failure.

Make missingness visible. A missing sample, dead sensor, dropped span, absent label, stale model, unreadable image, or lost symbol must not become zero, negative, or normal.

Shorten the path from mechanism to detection. Assertions, quality checks, embedded monitors, early clinical review, in-process inspection, and causal tests reduce the state that can be damaged or misinterpreted.

Design containment and safe degradation. Isolation, abstention, feature flags, page retirement, restricted operation, staged rollout, and escalation limit consequence while preserving comparative evidence.

Test the diagnostic system. Calibrate instruments, validate data paths, inject known faults, randomize explanatory models, audit thresholds, and verify that observers fail visibly.

Design within an explicit diagnostic budget. Coverage, capture depth, retention, inspection interval, and test selection are constrained by time, money, compute, storage, access, privacy, and risk. The budget and the evidence it excludes should be visible.

Close the learning loop. Findings should update design, standards, training, instrumentation, maintenance, workflow, datasets, models, and governance—not merely close the case.

25. A cross-strand evidence taxonomy

Table 7. Evidence categories that recur across the five histories.

CATEGORY	WHAT IT ESTABLISHES	EXAMPLES ACROSS STRANDS
Identity and context	What object, version, environment, and operating state the claim concerns	Patient/specimen; serial/slot; build/configuration; model/prompt; asset/location/load
Raw observation	The least-transformed available record	Narrative, waveform, image, dump, packet, sensor sample, radiograph, scan, fracture surface
Derived feature	A measured or computed property extracted from raw evidence	Interval, spectrum, stack, span duration, attribution, flaw size, health indicator
Expectation / model	What should happen under stated assumptions	Physiology, circuit or component model, specification, statistical model, finite-element or degradation model
Temporal evidence	How state and intervention changed through time	Symptom course, event trace, error trend, learning curve, agent trajectory, crack-growth history
Comparative evidence	How the case differs from baseline, peers, controls, or cohorts	Reference interval, known-good unit, healthy service, holdout set, fleet or structural peer
Intervention evidence	Whether changing a factor changes the predicted outcome	Treatment response, substitution, replay, rollback, ablation, controlled load, excavation
Outcome and consequence	What the state did to function, safety, cost, or mission	Clinical outcome, service impact, containment, model harm, structural capacity, recurrence
Observation-path health	Whether the evidence system itself was functioning	Quality controls, calibration, loss counters, sensor self-test, schema/version checks, evaluator validation
Governance and accountability	Who authorized, interpreted, acted, and reviewed	Consent, audit trail, change record, incident role, approval, inspection qualification, model card
Evidence integrity and adversarial resistance	Whether evidence is authentic, complete, and resistant to deliberate manipulation	Specimen chain of custody; signed logs; protected calibration records; poisoning checks; tamper-evident inspection data
Resource and capture budget	What coverage, fidelity, retention, and latency the investigation can afford	Test cost and access; telemetry sampling; compute and storage; inspection interval; escalation criteria

This taxonomy is intentionally plural. It resists the temptation to turn every artifact into a feature vector or every conclusion into one health score. Some evidence is valuable because it preserves a human account; some because it is physically close to the mechanism; some because it covers a population; some because it records who made a decision. A general diagnostic platform should join these categories without erasing their origin or authority. It also separates evidence availability from evidence integrity: an observation path can be silent by accident or persuasive by design after deliberate tampering. Resource and capture budget is included because every strand rations evidence, and those exclusions shape what can later be inferred.

Evidence strength depends on independence, relevance, and traceability rather than volume. Ten dashboards derived from one broken sensor do not provide ten confirmations. A single controlled intervention may discriminate more strongly than a million correlated observations. Conversely, one vivid artifact may be incidental unless it aligns with system behavior and consequence.

26. Analysis patterns as a candidate unifying layer

Proposal, not completed cross-domain standard. Pattern-Oriented Diagnostics and Pattern-Oriented Observability already provide a developed software-diagnostic lineage at DumpAnalysis.org. What remains provisional is their unification across all five strands. This synthesis treats this as a Pattern-Oriented Projection: recurring diagnostic operations are mapped across strands while medical, physical, computational, statistical, and institutional meanings remain intact. Appendix C tests three mappings end to end rather than treating a shared label as proof of transfer. [20–24]

Why analysis patterns rather than one universal ontology? A disease, a cracked weld, a marginal memory cell, a race condition, and a drifting model are not the same kind of thing. The analytical moves used to investigate them can nevertheless recur: establish a baseline, reconstruct identity and sequence, correlate independent observations, generate alternatives, perform a discriminating intervention, diagnose the observation path, and verify the effect of action. Analysis patterns unify the work performed on evidence rather than the objects being diagnosed.

Patterns as active tools. Problem patterns name recurrent diagnostic situations or evidence configurations; analysis patterns in this manuscript describe reusable transformations and investigative operations performed on evidence; implementation patterns realize those operations in a particular instrument or workflow; presentation patterns make results inspectable and contestable. This use of “analysis pattern” differs from Martin Fowler’s established usage for reusable conceptual domain models. The collision is acknowledged explicitly: Fowler models recurring business concepts, whereas the present proposal names recurring operations of diagnostic inquiry. [25–27]

Existing lineage at DumpAnalysis.org. Pattern-Oriented Diagnostics and Observability distinguishes problem patterns from analysis patterns and treats the latter as the grammar through which diagnostic artifacts are selected, transformed, compared, and interpreted. The four-part Pattern-Oriented Observability sequence develops observability as a component of the same pattern-oriented and systemic diagnostic programme [20–24].

- Part 1 - memory as the base slab beneath metrics, logs, and traces, with memory state and emitted messages related through analysis patterns and a memory-trace duality [21].
- Part 2 - a semiotics of memory, traces, and logs in which icons, indexes, Iconic Traces, and the Dia|gram language represent structure, behavior, observation, and interpretation [22].
- Part 3 - observation spaces for memory, traces, logs, metrics, narratives, states, and data, with analysis patterns serving as ways to compare and navigate diagnostic spaces [23].
- Part 4 - the Diagnostics–Observability Square, which distinguishes observability and diagnosability as properties from observation and diagnostics as processes, and names pattern-oriented observability as part of pattern-oriented and systemic diagnostics [24].

Table 8. Proposed cross-strand diagnostic-analysis-pattern families. The third column always follows this order: Medical; Hardware; Software; ML/AI; Technical. The labels are a synthesis rather than a claim that a formal catalogue already exists in every domain.

ANALYSIS-PATTERN FAMILY	REUSABLE OPERATION	PATTERN-ORIENTED PROJECTION ACROSS THE FIVE STRANDS
Baseline Comparison	Compare a case with a qualified prior state, peer, control, specification, or model.	Personal trajectory; known-good unit; healthy service cohort; holdout or reference distribution; qualified structural baseline.
Identity and Provenance Reconstruction	Rebuild the chain from diagnostic object through acquisition, transformation, and interpretation.	Patient and specimen; component, slot, and firmware; build and symbols; data, model, prompt, and evaluator; asset, location, procedure, and calibration.
Temporal Reconstruction	Order events, states, interventions, and gaps so that transition and mechanism can be examined.	Clinical course; first-failure record; execution or distributed trace; learning curve or agent trajectory; load and damage history.
Multimodal Correlation	Join evidence from partly independent modalities while preserving disagreement and missingness.	Examination, laboratory, and imaging; waveform, syndrome, and microscopy; dump, log, metric, trace, and profile; evaluation, attribution, and feedback; NDE, load, and fractography.
Differential Hypothesis Construction	Generate and rank competing states or mechanisms without turning a ranking into a verdict.	Differential diagnosis; candidate physical faults; software causal hypotheses; model, data, and pipeline failure classes; degradation mechanisms.
Causal Isolation by Intervention	Change one bounded factor and test whether the predicted observation changes.	Treatment response; component substitution or margining; replay, rollback, or minimized input; ablation or counterfactual test; controlled load, targeted inspection, or teardown.
Observation-Path Diagnosis	Test whether the sensor, specimen path, probe, telemetry pipeline, reconstruction, or evaluator is itself failing.	Quality controls; calibration and self-test; loss counters and symbol checks; judge and explanation sanity checks; sensor drift and inspection qualification.
Containment and Verification	Limit consequence, preserve evidence, and confirm that the intervention changes the expected outcome without unacceptable new harm.	Safety-netting or follow-up; fault containment and degraded mode; feature flag or rollback; abstention and staged release; restricted operation, repair, and reinspection.

Application to agentic AI diagnostics. The memory-analysis and trace-and-log-analysis pattern languages are applicable to agentic AI when translated to the agent as a diagnostic object. Context windows, persistent memories, scratchpads, retrieved documents, checkpoints, and tool state can be examined as memory snapshots or memory structures. Prompts, planning steps, tool calls, observations, retries, hand-offs, memory updates, and external side effects form traces and logs, often with partial rather than total order. Patterns for missing or inconsistent state, incomplete history, bifurcation, semantic mapping, inter-correlation, repeated activity, discontinuity, and observation-path failure can therefore guide analysis of why an agent lost context, looped, diverged, selected an unsuitable tool, or caused an unsafe side effect [4, 21–24].

Memory and trajectory must be read together. A memory view asks what context and state existed at a selected step; a trace-and-log view asks how that state was formed and which action followed. Their cooperation reuses the memory-trace relationship described in Pattern-Oriented Observability and aligns it with the ML and AI history's emphasis on prompts, retrieval, memory, tools, trajectories, and side effects. The claim concerns externally observable and preserved agent behavior; it should not be presented as direct access to hidden reasoning or as proof that every internal representation has a stable human meaning [4, 21–24].

A portable diagnostic-analysis-pattern description should preserve more than a name. At minimum it should state Intent, Context, Problem, Required Evidence, Transformation or Procedure, Alternatives and Counter-evidence, Observer Effects and Risks, Resulting Context, Verification, Stopping Conditions, and Known Uses. Known Uses records recurrence evidence; Stopping Conditions state when further inquiry is no longer justified. The same abstract pattern may then be implemented as a clinical workflow, inspection procedure, debugger command sequence, telemetry query, evaluation protocol, checklist, or agent tool without claiming identical consequences.

A middle layer for people and machines. Analysis patterns can sit between domain-specific evidence and general-purpose automation. Below them remain the ontologies, instruments, standards, physics, ethics, and decision rights of each strand. Above them, investigators and AI systems can compose reusable operations into a transparent diagnostic plan. The pattern name should select a method of inquiry, not certify its conclusion; every invocation must still expose its inputs, assumptions, transformations, alternatives, and stopping conditions.

Limits of the unification. Pattern similarity is not causal identity. A pattern transferred from software to medicine or from hardware to technical inspection must be requalified for the domain, including what can be measured, what intervention is permissible, who may authorize it, and what harm an error can cause. Used with this constraint, analysis patterns offer a common language for diagnostic reasoning without collapsing the five strands into one discipline.

27. Comparison with other unifiers and what would refute the proposal

Analysis patterns are one candidate unifier, not the only or presumptively strongest one. Established alternatives unify different layers of diagnostic work. Model-based diagnosis generates candidates from inconsistency between observations and a structural component model; fault detection and isolation generates residuals and isolates faults in dynamic engineering systems; Bayesian decision theory updates probabilities under asymmetric loss; control-theoretic observability formalizes whether internal state can be inferred from outputs; FMEA and fault trees organize anticipated failure modes and causal combinations; semiotics organizes relations among signs, objects, and interpretations. [6, 8, 14, 28–31]

Table 9. Candidate cross-strand unifiers: scope, requirements, strengths, and limits.

CANDIDATE	WHAT IT UNIFIES	REQUIRES	STRENGTH	LIMIT ACROSS FIVE STRANDS
Diagnostic analysis patterns	Operations performed on evidence	Pattern description, domain qualification, known uses	Transfers inquiry without requiring one ontology	Recurrence and benefit must be demonstrated; names can become relabelling
Model-based diagnosis	Candidate generation from inconsistency	Structural model, component modes, observations	Formal and discriminating where models exist	Weak where structure or failure modes cannot be specified credibly
Fault detection and isolation	Residual generation and fault isolation	Dynamic process or signal model, sensors, thresholds	Strong for machinery, vehicles, plants, and control systems	Limited for narrative, ethical, institutional, and open-ended diagnostic objects
Bayesian decision theory	Belief updating and action under loss	Probabilities or likelihoods, outcomes, loss function	Genuinely cross-domain and explicit about thresholds	Does not reconstruct identity, sequence, mechanism, or authority by itself
Control-theoretic observability	Inferability of state from outputs	Specified dynamic state-space model	Formal criterion with clear mathematical meaning	Not directly portable to changing socio-technical systems without a defensible model

CANDIDATE	WHAT IT UNIFIES	REQUIRES	STRENGTH	LIMIT ACROSS FIVE STRANDS
FMEA and fault trees	Anticipated failure modes, effects, and causal combinations	System boundary, enumerated modes or top event	Widely transferred and useful before operation	Completeness is difficult; novel interactions and learned behaviour escape enumeration
Semiotics	Relations among signs, objects, and interpretations	Representational vocabulary and interpretive rules	Explains mediation and meaning across artifacts	Provides less direct guidance for test selection, action, and verification

The comparative claim is therefore narrower and more defensible: diagnostic analysis patterns may provide a useful inquiry-level layer above formal models and domain practices because they can describe what investigators do when no single structural, probabilistic, or control-theoretic formalism covers the object. Where one of those formalisms is available, the pattern should invoke it rather than compete with it.

What would refute this proposal. The proposal should be rejected or narrowed if any of the following occurs: (1) a proposed pattern family cannot state Required Evidence, Stopping Conditions, and Verification in one or more strands; (2) following the pattern produces materially worse investigation or action than established domain practice; (3) independent practitioners judge that the mapping merely renames different operations; (4) worked cases fail to preserve a recognisable operation after domain-specific qualification; or (5) the pattern adds no discriminating evidence, safe action, or transferable learning. Appendix C tests only three families, so the present synthesis establishes limited feasibility, not comprehensive validation.

Reflexive application to this manuscript. Version 9 failed its own principles because its citation graph was closed, its candidate unifier was not invoked in a worked case, its continuum was inconsistent, and its sources were not identified with the exactness demanded of diagnostic evidence. The present synthesis treats those failures as observations about the proposal's observation path and revises the evidence base, demonstrations, comparison set, falsification conditions, and provenance disclosure accordingly.

28. AI as instrument, subject, and governor

Analysis patterns provide a possible action vocabulary for an evidence-oriented copilot, but they do not make its conclusions self-validating. AI occupies three positions in the synthesis. As an instrument, it detects patterns, classifies images, retrieves similar cases, ranks hypotheses, predicts degradation, and operates diagnostic tools. As a subject, it requires diagnosis of data, training, model internals, serving, feedback, evaluation, and agent behavior. As a governor, it may recommend or execute actions that change systems, patients, assets, and the evidence used for its own future learning.

Agentic AI makes the pattern-language bridge concrete. Memory analysis patterns can structure preserved context, persistent memory, retrieved evidence, and tool state, while trace and log analysis patterns can structure the ordered or partially ordered trajectory of instructions, plans, tool calls, observations, retries, hand-offs, and side effects. Together they support diagnosis across state and sequence: what the agent retained, how that state arose, what it did next, and where the observation path may be incomplete or misleading [4, 21–24].

The third role creates a recursive assurance problem. An AI system that removes a server from service, changes a maintenance interval, prioritizes a patient, blocks a transaction, or rewrites a configuration alters both the world and the dataset by which it will later be judged. Diagnostic design should therefore preserve counterfactuals, protected comparisons, audit, human review, and the distinction between recommendation and authority.

The most credible architecture is not an autonomous oracle but an evidence-oriented copilot. It should maintain a case file, cite artifacts, run reproducible queries, track alternatives, identify missing evidence, and propose safe discriminating tests. Its success should be evaluated by causal correctness, evidence quality, safety, and reduced recurrence—not by the fluency of its explanation or agreement with one historical label. Human review is not assumed to be reliable merely because a person is present: automation bias, monitoring complacency, alert fatigue, skill decay, and production pressure can convert nominal oversight into routine confirmation. [32–34]

Oversight should therefore be observable and testable. Controls include recording an independent judgement before revealing the automated recommendation where feasible; sampling accepted recommendations for audit; measuring override, escalation, and disagreement rates; rotating staff through unaided practice; designing alerts around action rather than volume; and monitoring the human–automation team for drift. Responsibility remains human and institutional, but performance must be evidenced rather than presumed.

29. Reflexive diagnostics: diagnosing observability, diagnostics, and debugging

A mature diagnostic discipline eventually turns back upon itself. It must diagnose not only the patient, device, program, model, or asset, but also the observability system, diagnostic workflow, debugging practice, evidence pipeline, and explanatory instruments through which that object becomes knowable. Missing telemetry, broken identity, distorted baselines, misleading dashboards, hallucinated AI explanations, unsafe probing, and unvalidated evaluators are failures of diagnosis itself. This creates a second-order layer in which diagnostics, observability, and debugging become diagnostic objects. [35]

Pattern-Oriented Observability already points in this direction through observation-path diagnosis, the Diagnostics–Observability Square, and explicit attention to observer failure [21–24]. The same logic extends beyond observability alone. A debugging intervention can destroy timing evidence; a dashboard can normalize a failing baseline; an automated explanation can become persuasive while unsupported; and a diagnostic agent can modify the very state by which it will later be judged. Reflexive diagnosis names the disciplined observation, diagnosis, and debugging of these diagnostic means themselves.

This second-order perspective recurs across all five strands. Medicine must diagnose tests, workflows, triage pathways, and decision support. Hardware practice must validate probes, error-reporting channels, BIST coverage, and service records. Software practice debugs logs, traces, metrics, profilers, and incident procedures. ML and AI practice evaluates evaluators, explanation methods, drift detectors, and agentic diagnostic assistants. Technical diagnostics must qualify sensors, NDE procedures, structural-health-monitoring pipelines, and digital twins. In every case the question is not only What is wrong with the object? but also Are our means of seeing, explaining, and correcting it themselves adequate? The human overseer is also part of the diagnostic path and must be assessed for automation bias, fatigue, deskilling, handover loss, and conflicting incentives. [32, 33, 36]

Reflexive diagnosis is therefore not an optional philosophical add-on. It is an operational requirement for trustworthy observability, diagnostics, and debugging. Mature diagnostic systems should expose the health of their sensors, telemetry paths, evaluators, reconstruction steps, and automated reasoning components; preserve what was missing, filtered, or overridden; and support safe debugging of the diagnostic process itself. The result is a recursive discipline in which systems of observation, interpretation, and correction become diagnostic objects in their own right. Deliberate manipulation must be distinguished from accidental failure: specimen substitution, falsified inspection records, signed-log failure or log tampering, data poisoning, and evaluation gaming can correlate supposedly independent evidence paths. Integrity, authentication, sequencing, and adversary-aware validation are therefore reflexive diagnostic requirements. [37, 38]

30. Pattern narratives and narratological unification

Analysis patterns provide reusable local forms of inquiry, but real investigations usually unfold as compositions of such forms. A clinician, failure analyst, debugger, or ML investigator rarely applies only one pattern. Instead the case develops through a narrated sequence: anomaly recognition, evidence preservation, contextualization, comparative assessment, hypothesis construction, discriminating intervention, and outcome verification. The resulting explanation is not merely a label; it is a pattern-guided account of how the observed signs came to be.

Higher-Order Pattern Narratives (Analysing Diagnostic Analysis) makes the compositional and reflexive point explicit: diagnosis is often not one pattern but a sequence of patterns joined by time, evidence transformation, and explanatory intent; the diagnostic analysis can itself become the object of another analysis. The linked narratological extension, Unified Computer Diagnostics: Incorporating Hardware Narratology, shows how hardware and software evidence can be reconstructed as one constrained cross-layer story. [35, 39]

This narrative layer is particularly useful for agentic AI diagnostics. An agentic case rarely reduces to a single prompt-output pair. It unfolds through remembered state, retrieved context, planned steps, tool calls, observations, retries, hand-offs, memory updates, and external side effects. Memory-analysis patterns and trace-and-log-analysis patterns therefore supply reusable local readings, while pattern narratives connect them into a trajectory that can be inspected, challenged, and improved. Recorded behavior is still not direct access to hidden reasoning, but it is a structured basis for accountable diagnosis.

Narratology does not make diagnosis fictional. It names the disciplined reconstruction of a sequence, mechanism, and consequence from constrained evidence. A good diagnostic narrative preserves provenance, distinguishes observed events from inferred transitions, records missingness, and remains open to revision under disconfirming evidence. In this sense, pattern narratives complement analysis patterns: the patterns provide reusable local moves, while the narrative organizes them into a coherent, testable case account.

PART VII

Future directions and conclusion

31. Interoperable evidence ecosystems

The future of diagnostics is not one universal database but an interoperable evidence ecosystem. Images, waveforms, dumps, traces, specimens, models, maintenance records, outcomes, and narratives should retain domain-specific structure while sharing enough identity, time, provenance, and semantics to be joined. DICOM and DICONDE, OpenTelemetry, laboratory identifiers, model registries, digital threads, and asset genealogies are partial examples of this direction [1–5].

Evidence graphs may become a common analytical layer. Nodes can represent observations, states, versions, components, hypotheses, interventions, and outcomes; edges can distinguish physical connection, data derivation, temporal order, dependency, ownership, and inferred causality. Such graphs are useful only when fact and inference remain typed, confidence is explicit, and every transformation can be inspected.

Adaptive capture will also grow. Systems can preserve broad low-cost signals continuously, then deepen capture around novelty, risk, or anomaly: a richer dump, higher-rate waveform, targeted trace, repeated specimen, focused scan, or preserved agent trajectory. This is an explicit optimisation under a diagnostic budget: sampling, retention, inspection frequency, compute, storage, privacy, and operator time determine coverage. The challenge is to preserve enough prehistory before the anomaly is recognised while making the excluded evidence and sampling policy visible. [40]

32. Governed diagnostic agents and digital twins

Digital twins and diagnostic agents converge around a maintained model of a particular object. A useful twin is not merely a visualization; it binds identity, configuration, observed state, model assumptions, uncertainty, and a decision scope. A useful agent is not merely a chatbot; it binds tools, permissions, evidence, hypotheses, actions, and an audit trail. Both become dangerous when the representation silently drifts away from the counterpart or when action exceeds validation.

Future agents should be designed like safety-critical diagnostic instruments. Read-only access should be the default. Mutation should require scoped credentials, bounded commands, reversible actions, and approval proportional to consequence. The agent should record which query produced which observation, distinguish retrieved evidence from generated inference, test counter-hypotheses, and stop when qualification or coverage is exceeded.

Across the five strands, the technical frontier is self-observation without self-certification. Sensors should report their own health; telemetry pipelines should expose loss; models should detect domain shift; clinical and engineering workflows should reveal unresolved results; twins should compare prediction with outcome; agents should surface when their tools, evidence, or authority are insufficient. A system that declares itself healthy without independent checks has merely automated reassurance.

33. Conclusion

The five histories converge on a larger story: the human effort to make hidden condition observable and to act responsibly on what is learned. Medicine makes illness and care trajectories legible; hardware makes physical faults and encoded errors traceable; software makes execution and distributed causality reconstructable; ML and AI make learned behavior, data dependence, and agent trajectories contestable; technical diagnostics makes the integrity and degradation of physical assets assessable [1–5].

None of the strands advances by replacing earlier evidence. Bedside observation survives beside genomic testing; oscilloscopes survive beside fleet telemetry; dumps survive beside distributed traces; residuals survive beside mechanistic interpretability; visual inspection survives beside structural-health monitoring. Progress comes from layering views, preserving their conditions, and creating stronger ways to discriminate among explanations.

A unified diagnostic discipline should therefore remain modest in claim and ambitious in design. It should capture before evidence disappears, preserve identity and provenance, compare against qualified references, separate observation from inference and action, combine independent modalities, expose missingness, test its own instruments, and connect intervention to verified consequence. AI can accelerate this work, but it cannot relieve people and institutions of judgment, ethics, or accountability.

Analysis patterns offer one possible bridge between the shared continuum and domain practice. This bridge builds on the existing Pattern-Oriented Diagnostics and Pattern-Oriented Observability lineage at DumpAnalysis.org [20–24]. Its memory and trace-and-log pattern languages can also be applied to agentic AI diagnostics by joining preserved context and memory state with trajectories of prompts, tool calls, observations, retries, and side effects. Pattern narratives extend this bridge by showing how reusable local operations combine into evidential stories of anomaly, context reconstruction, hypothesis formation, discriminating intervention, and verification [35, 39]. The patterns make evidence operations teachable, composable, and invocable, while their implementations remain accountable to the distinct physics, semantics, ethics, and consequences of each strand. This is a proposal for unification at the level of inquiry, not a claim that all diagnostic objects or explanations are equivalent.

The durable synthesis is the canonical continuum used throughout this synthesis: Observe; Measure / preserve; Contextualize; Compare; Infer; Discriminate; Act; Verify and learn. Each strand gives those stages a different object and moral weight. Together they show that observability is not the end of diagnosis, and diagnosis is not the end of responsibility. The final achievement is an explanation that can be challenged, an action that can be justified, and a system that becomes more diagnosable because the investigation occurred.

APPENDIX A

Comparative matrix

Table 10. A compact comparison of the five diagnostic strands.

STRAND	PRIMARY QUESTION	CORE EVIDENCE	DISTINCTIVE LIMIT / DUTY
Medical	What condition best explains this person’s evidence, and what should care do next?	Narrative, examination, specimens, images, signals, population evidence, follow-up	Reference standards are imperfect; patient agency, communication, equity, and clinical utility are integral
Hardware	Which physical or design fault produced the observed error or service failure?	Waveforms, checks, syndromes, counters, firmware records, topology, returned-part analysis	Correction and replacement can hide mechanism; observer and management paths must be trusted
Software	What happened in execution, why, and which change will prevent recurrence?	Code, configuration, dumps, logs, metrics, traces, profiles, deployment and incident records	State is volatile and distributed; instrumentation and remediation can perturb the system
ML and AI	Why did learned behavior fail under this data, task, population, or trajectory?	Residuals, splits, gradients, features, attributions, evaluations, drift, feedback, agent actions	Behavior can fail without a software error; metrics and explanations are conditional and must be tested
Technical	What damage, degradation, or integrity state best explains the physical observations?	Inspection, NDE, strain, vibration, thermography, material analysis, loads, condition histories	Detectability, access, qualification, and structural consequence bound every conclusion

APPENDIX B

Shared glossary

Working cross-strand definitions. Domain standards may use narrower meanings.

Abductive reasoning. Inference from observations to one or more plausible explanatory hypotheses, followed by comparison and testing rather than treatment as a deduction.

Abstention. A deliberate decision not to classify, predict, diagnose, or act because evidence, qualification, or authority is insufficient.

Analysis pattern. A reusable transformation, comparison, or interpretive operation that makes diagnostic evidence intelligible and supports recognition or testing of problem patterns.

Anomaly. A departure from a qualified baseline, model, distribution, peer population, or expected relationship.

Automation bias. A tendency to over-rely on automated recommendations or to under-detect automation errors, especially when automation is usually reliable.

Baseline. A reference condition, population, model, version, or period against which later evidence is compared.

Calibration. The relation between an instrument, probability, or score and a reference or observed outcome under stated conditions.

Causal isolation. Discrimination among mechanisms that can produce similar symptoms, usually through independent evidence or controlled intervention.

Consequence assessment. Evaluation of what an identified state could do to health, function, safety, mission, environment, cost, or rights.

Context. Identity, configuration, environment, operating state, population, purpose, and time needed to interpret an observation.

Coverage. The portion of relevant states, faults, cases, paths, populations, or physical regions that an observation or test can examine.

Debugging. Controlled investigation of where and how behavior departs from intent, often by stopping, stimulating, substituting, varying, or replaying.

Diagnosis. Evidence-based identification and explanation of the state or mechanism that best accounts for observations.

Diagnostic object. The entity and boundary to which a diagnostic claim applies: person, component, execution, model system, asset, or organization.

Digital twin. A maintained digital representation of an identified counterpart, updated with evidence for a declared diagnostic or predictive use.

Drift. Change over time in an instrument, population, data distribution, model relationship, process, or baseline.

Evidence. An observation or preserved artifact whose production path and relevance allow it to support or challenge a claim.

Evidence graph. A structure linking observations, states, versions, components, hypotheses, interventions, and outcomes with typed relationships.

Expected loss. The probability-weighted consequence of possible outcomes under a decision, used to compare tests and actions when errors have asymmetric costs.

Fault. An abnormal physical, design, data, model, or causal condition capable of producing error, damage, or failure.

Feature. A measured or computed property extracted from raw evidence for comparison, classification, estimation, or diagnosis.

Intervention. A deliberate change in treatment, load, configuration, component, input, model, route, or policy used to discriminate or correct.

Known Uses. Documented cases in which a pattern has been applied, providing evidence of recurrence, transfer, and limitations.

Likelihood ratio. The probability of a test result under the target condition divided by its probability under an alternative condition; it updates prior odds to posterior odds.

Missingness. Evidence that is absent, lost, uncollected, uninterpretable, or unavailable; it must not be represented as a negative or normal observation.

Monitoring. Repeated observation of selected conditions and rules to identify change requiring attention or action.

Observability. The designed capacity to infer relevant hidden behavior or state from obtainable outputs and preserved context.

Observation path. The sensors, specimen handling, acquisition, software, transformations, communication, and interpretation through which evidence is produced.

Pattern-oriented diagnostics. A diagnostic discipline that uses explicit repertoires of problem, analysis, implementation, and presentation patterns to structure evidence, interpretation, intervention, and learning.

Pattern-oriented observability. The observability component of pattern-oriented and systemic diagnostics, concerned with how observations are engineered, acquired, represented, related, and interpreted through patterns.

Pattern narrative. A structured composition of diagnostic patterns into an evidence-based explanatory sequence linking anomaly, context, hypothesis, intervention, and verification.

Prognosis. A conditional estimate of future condition, outcome, failure probability, or remaining useful life under stated assumptions.

Provenance. The documented origin and transformation history of evidence, including identity, method, version, processing, and responsible operations.

Reference standard. The method or composite evidence used to establish the target state for comparison; no reference is universally infallible.

Reflexive diagnostics. The observation, diagnosis, and debugging of the diagnostic, observability, or debugging system itself, including sensors, telemetry paths, evaluators, workflows, and automated assistants.

Residual. A discrepancy between observed behavior and the value predicted by a model or expected relation.

Root cause. A causal mechanism or contributing condition at a stated level of analysis; complex systems may have several legitimate causal levels.

Threshold. A decision boundary applied to a measurement, feature, probability, score, or risk estimate.

Trace. An ordered record of selected events, operations, states, or actions used to reconstruct behavior over time.

Uncertainty. Quantified or stated doubt about observation, inference, coverage, model, consequence, or future condition.

Backpropagation telemetry. Monitoring of gradients, activations, losses, parameter updates, and numerical conditions during model training.

Diagnosability. The degree to which available observations and tests can distinguish relevant fault states or explanations; unlike observability, it concerns discrimination among diagnostic alternatives.

Dia|gram language. A visual notation used in Pattern-Oriented Observability to represent structures, events, relations, and transformations across memory, trace, and log evidence.

First-failure data capture. Preservation of the earliest trustworthy evidence before retry, reset, failover, repair, or replacement changes system state.

Fitness-for-service. A structured engineering assessment of whether damaged or degraded equipment can continue operation under stated conditions and for a stated period.

Iconic Trace. A diagrammatic trace representation in which selected events and relations are expressed through visual icons and spatial organization.

Implementation pattern. A recurring way to realize a diagnostic or analysis operation in a particular tool, instrument, platform, or workflow.

Memory–trace duality. The relationship between state preserved at a point and the event history through which that state was formed; each constrains interpretation of the other.

Post-test probability. The probability of a target condition after current test evidence has been combined with pre-test probability.

Pre-test probability. The estimated probability of a target condition before the current test, based on context, prevalence, and prior evidence.

Prevalence. The proportion of a defined population that has the target condition in a stated setting and period; it influences predictive values.

Presentation pattern. A recurring way to render diagnostic evidence or conclusions so that relationships, uncertainty, and provenance remain inspectable.

Probability of detection. The probability that a specified inspection system will detect a flaw of a given size or condition, usually estimated with uncertainty bounds.

Problem pattern. A recurrent diagnostic situation, evidence configuration, or failure manifestation that calls for analysis.

Prognostics. The discipline of estimating future condition, failure probability, or remaining useful life under stated assumptions.

Shortcut learning. Use by a learned model of an easy but non-general causal or correlational cue instead of the intended task-relevant relationship.

Sparse feature. A learned or engineered feature that is active for relatively few inputs or dimensions; in interpretability work, such features may be derived from sparse representations.

Stopping condition. A criterion for ending or narrowing an investigation because further expansion is unlikely to change action, responsibility, prevention, or evidential confidence enough to justify its cost.

Training–serving skew. A mismatch between data, features, preprocessing, or conditions used during model training and those encountered during deployment.

Validation. Evidence that a method, model, instrument, workflow, or system is adequate for its intended real-world use.

APPENDIX C

Worked cross-strand pattern cases

These cases test whether a diagnostic-analysis-pattern family transfers as an operation rather than merely as a label. Each case follows the expanded pattern template and states where the analogy breaks. They are demonstrations of limited feasibility, not proof that every family transfers to every strand.

Worked case 1. Observation-Path Diagnosis: NDE sensor drift and a dropped-span telemetry pipeline

Intent. Determine whether an apparent change in the diagnosed object is instead produced by the observation path.

Table 11. Observation-Path Diagnosis: NDE sensor drift and a dropped-span telemetry pipeline.

FIELD	Technical diagnostics: ultrasonic/NDE channel	Software diagnostics: distributed trace
Context	A structural-health or NDE channel reports increasing indication amplitude on an asset whose load regime appears unchanged.	A service trace shows a missing downstream span during an incident.
Required Evidence	Asset and sensor identity; calibration and reference-block results; coupling, temperature, procedure, operator, raw waveform, prior inspections, and independent modality.	Trace and span identifiers; propagation headers; instrumentation and collector versions; sampling policy; export and drop counters; service logs and metrics.

FIELD	Technical diagnostics: ultrasonic/NDE channel	Software diagnostics: distributed trace
Transformation / Procedure	Re-run reference standard; compare raw and derived signals; swap or independently calibrate sensor; repeat under controlled geometry; compare with another method.	Inspect context propagation; compare logs and metrics for the supposed call; check sampler and collector loss; reproduce with controlled instrumentation; inspect an unsampled or locally captured request.
Alternatives and Counter-evidence	Real damage growth, load change, coupling change, material or temperature effect, processing change.	The downstream call never occurred; asynchronous work lost parentage; instrumentation was absent; the service failed before span creation.
Observer Effects and Risks	Repeated loading or invasive access may extend damage; recalibration can overwrite the failing state.	Extra tracing can change timing, cost, and cardinality; enabling full capture may expose sensitive data.
Stopping Conditions	Stop observation-path inquiry when independent calibration and modality agree that the channel is healthy, or when correcting the channel removes the anomaly and the asset result returns to baseline.	Stop when propagation, sampling, and collection are verified for a reproduced request, or when repairing the telemetry path restores the missing causal relation without changing service behaviour.
Verification	A known reference is measured correctly; an independent channel agrees; subsequent asset trend remains coherent.	Controlled requests yield complete linked spans; drop counters remain zero or quantified; incident interpretation agrees with logs and metrics.
Known Uses	Calibration checks, probability-of-detection qualification, sensor plausibility, repeat inspection.	Trace-context validation, instrumentation health checks, collector monitoring, loss and sampling audits.
Where the analogy breaks	The sensor interacts with material, geometry, coupling, and physical access; inspection can alter the asset.	The path is symbolic and software-defined; causality may be partial, and instrumentation can be changed without touching the physical service.

Evidence base: [10, 11, 15, 19, 40].

Worked case 2. Baseline Comparison: a personal medical trajectory and an ML holdout distribution

Intent. Compare current evidence with a qualified reference while preserving the reference's population, time, method, and decision purpose.

Table 12. Baseline Comparison: a personal medical trajectory and an ML holdout distribution.

FIELD	Medical diagnostics: within-person biomarker trend	ML/AI diagnostics: production distribution versus holdout
Context	A patient's biomarker rises but remains inside a broad population reference interval.	A deployed classifier retains overall accuracy while confidence and subgroup performance change.
Required Evidence	Repeated results from comparable methods; specimen conditions; treatment and symptom timeline; population interval; analytical variation; intended clinical decision.	Training and holdout definitions; feature and label distributions; model and pipeline versions; calibration and subgroup metrics; serving inputs and outcomes.
Transformation / Procedure	Estimate within-person change relative to analytical and biological variation; compare with population interval; update pre-test probability and consequences.	Compare production and holdout distributions; stratify cohorts; examine calibration and outcome frequencies; test whether drift is data, label, pipeline, or model related.
Alternatives and Counter-evidence	Method change, specimen handling, transient physiology, treatment effect, unrelated comorbidity, population interval inappropriate to the person.	Logging or labelling change, delayed outcomes, selection bias, feature pipeline defect, seasonal change, benign covariate shift.
Observer Effects and Risks	Repeat testing can trigger cascades, anxiety, cost, or treatment changes that alter the trajectory.	Monitoring choices can leak labels, influence retraining data, or optimise only measured cohorts.
Stopping Conditions	Stop when the available evidence changes or confirms the clinical action and further testing has lower expected value than follow-up or treatment.	Stop when the responsible layer is isolated sufficiently to choose rollback, recalibration, retraining, abstention, or continued monitored operation.
Verification	Follow-up, orthogonal test, treatment response, or outcome behaves as predicted.	Prospective production outcomes and calibration improve after the chosen intervention without unacceptable subgroup harm.

FIELD	Medical diagnostics: within-person biomarker trend	ML/AI diagnostics: production distribution versus holdout
Known Uses	Serial biomarkers, personal baselines, delta checks, repeated imaging and vital signs.	Drift detection, calibration monitoring, holdout and shadow evaluation, subgroup surveillance.
Where the analogy breaks	The reference concerns a person with agency and care consequences; “normal” is not purely statistical.	The reference distribution is engineered and can be regenerated, but labels and environments may change after deployment.

Evidence base: [7, 8, 12, 13].

Worked case 3. Temporal Reconstruction: a crash dump and an agent trajectory

Intent. Reconstruct the sequence that produced a final state by joining a snapshot with ordered or partially ordered history.

Table 13. Temporal Reconstruction: a crash dump and an agent trajectory.

FIELD	Software diagnostics: crash dump plus telemetry	Agentic AI diagnostics: memory state plus tool trajectory
Context	A process crashes in an allocator with a corrupted heap; the dump preserves the terminal state but not the earlier write.	An agent produces an unsafe side effect; its final response does not reveal how context, tools, and memory produced the action.
Required Evidence	Exact binaries and symbols; dump; thread and heap state; logs, traces, profiles, deployment and configuration history; clock and loss information.	Model and policy identifier; system and task instructions; retrieved context; memory checkpoints; tool calls and results; retries, hand-offs, timestamps, side effects, and missing-event markers.
Transformation / Procedure	Reconstruct identity and terminal state; align prior events; search backward from corrupted structures; compare healthy runs; replay or instrument a minimized reproduction.	Join memory snapshots to trajectory events; establish partial order and causal links; locate context loss, loop, unsuitable tool choice, or policy failure; replay safely where possible.
Alternatives and Counter-evidence	Allocator defect, earlier overwrite, symbol mismatch, dump truncation, observer-induced schedule change.	Tool error, stale retrieval, policy misconfiguration, memory overwrite, missing trace step, external environment change, misleading generated explanation.

FIELD	Software diagnostics: crash dump plus telemetry	Agentic AI diagnostics: memory state plus tool trajectory
Observer Effects and Risks	Instrumentation and replay can change scheduling; dumps contain sensitive memory.	Full trajectory capture can expose sensitive prompts and data; replay may repeat side effects; generated rationales are not ground truth.
Stopping Conditions	Stop when a reproducible transition explains the corrupted state and a corrective change prevents recurrence, while remaining alternatives are bounded.	Stop when the evidence-supported trajectory identifies a controllable failure point and a safe intervention changes the predicted outcome; do not claim hidden reasoning beyond recorded behaviour.
Verification	Regression test, deterministic replay, sanitizer, or production outcome confirms prevention.	Sandbox replay, counterfactual prompt or tool restriction, and monitored deployment confirm safer behaviour.
Known Uses	Post-mortem dump analysis, record-and-replay, distributed incident timelines.	Agent observability, tool-call traces, memory audits, trajectory evaluation, side-effect review.
Where the analogy breaks	Program execution may be deterministically replayable and grounded in exact code and machine state.	Generative behaviour may be stochastic; model internals and natural-language rationale do not provide a stable, complete causal account.

Evidence base: [1, 4, 21–23, 35].

References

References are numbered in true first-citation order. Live web sources were accessed on 2 August 2026 unless a different access date is stated. The five companion histories remain the historical source base; external primary papers, official standards, specifications, and authoritative guidance directly support the cross-strand claims introduced in this synthesis.

[1] History of Software Diagnostics, Observability, and Debugging. DumpAnalysis.org, July 2026. Accessed 2 August 2026.
[2] History of Hardware Diagnostics, Observability, and Debugging. DumpAnalysis.org, July 2026. Accessed 2 August 2026.
[3] History of Medical Diagnostics, Observability, and Debugging. DumpAnalysis.org, July 2026. Accessed 2 August 2026.
[4] History of ML and AI Diagnostics, Observability, and Debugging. DumpAnalysis.org, August 2026. Accessed 2 August 2026.
[5] History of Technical Diagnostics, Observability, and Debugging. DumpAnalysis.org, August 2026. Accessed 2 August 2026.
[6] Kalman, R. E. "On the General Theory of Control Systems." Proceedings of the First International Congress of the International Federation of Automatic Control, 1960, pp. 481–492. Accessed 2 August 2026.
[7] U.S. Food and Drug Administration. Biomarker Qualification: Evidentiary Framework. Draft Guidance for Industry and FDA Staff, December 2018. Accessed 2 August 2026.
[8] Fagan, T. J. "Nomogram for Bayes Theorem." New England Journal of Medicine 293, no. 5 (1975): 257. Accessed 2 August 2026.
[9] Avižienis, A., J.-C. Laprie, B. Randell, and C. Landwehr. "Basic Concepts and Taxonomy of Dependable and Secure Computing." IEEE Transactions on Dependable and Secure Computing 1, no. 1 (2004): 11–33. Accessed 2 August 2026.
[10] World Wide Web Consortium. Trace Context, W3C Recommendation, 23 November 2021. Accessed 2 August 2026.
[11] OpenTelemetry. OpenTelemetry Specification 1.59.0. Accessed 2 August 2026.
[12] Sculley, D., et al. "Hidden Technical Debt in Machine Learning Systems." Advances in Neural Information Processing Systems 28 (2015): 2503–2511. Accessed 2 August 2026.

[13] Guo, C., G. Pleiss, Y. Sun, and K. Q. Weinberger. "On Calibration of Modern Neural Networks." Proceedings of the 34th International Conference on Machine Learning, PMLR 70 (2017): 1321–1330. Accessed 2 August 2026.

[14] Isermann, R. "Fault Diagnosis of Machines via Parameter Estimation and Knowledge Processing—Tutorial Paper." Automatica 29, no. 4 (1993): 815–835. Accessed 2 August 2026.

[15] National Aeronautics and Space Administration. NASA-STD-5009C: Nondestructive Evaluation Requirements for Fracture Critical Metallic Components. 2023. Accessed 2 August 2026.

[16] API 579-1/ASME FFS-1. Fitness-For-Service. 4th ed., 2021. Accessed 2 August 2026.

[17] DICOM Standards Committee. Digital Imaging and Communications in Medicine (DICOM), current edition. Accessed 2 August 2026.

[18] ASTM International. ASTM E2339-21: Standard Practice for Digital Imaging and Communication in Nondestructive Evaluation (DICONDE). 2021. Accessed 2 August 2026.

[19] Parker, P. A., et al. Guidebook for the Design and Analysis of a NASA Standard Nondestructive Evaluation Probability of Detection Study. NASA/TM-20220013822, 2022. Accessed 2 August 2026.

[20] Pattern-Oriented Diagnostics and Observability as a Philosophy of Engineering. DumpAnalysis.org, August 2026. Accessed 2 August 2026.

[21] Pattern-Oriented Observability (Part 1): The Base Slab and Foundation of Observability Pillars. DumpAnalysis.org, August 2026. Accessed 2 August 2026.

[22] Pattern-Oriented Observability (Part 2): Semiotics of Memory, Trace, and Log Analysis. DumpAnalysis.org, August 2026. Accessed 2 August 2026.

[23] Pattern-Oriented Observability (Part 3): Observation Spaces. DumpAnalysis.org, August 2026. Accessed 2 August 2026.

[24] Pattern-Oriented Observability (Part 4): Diagnostics–Observability Square (DOS). DumpAnalysis.org, August 2026. Accessed 2 August 2026.

[25] Alexander, C., S. Ishikawa, M. Silverstein, et al. A Pattern Language: Towns, Buildings, Construction. New York: Oxford University Press, 1977. Accessed 2 August 2026.

[26] Gamma, E., R. Helm, R. Johnson, and J. Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software. Reading, MA: Addison-Wesley, 1995. Accessed 2 August 2026.

[27] Fowler, M. Analysis Patterns: Reusable Object Models. Reading, MA: Addison-Wesley, 1997. Accessed 2 August 2026.

[28] Reiter, R. "A Theory of Diagnosis from First Principles." Artificial Intelligence 32, no. 1 (1987): 57–95. Accessed 2 August 2026.

[29] de Kleer, J., and B. C. Williams. "Diagnosing Multiple Faults." Artificial Intelligence 32, no. 1 (1987): 97–130. Accessed 2 August 2026.

[30] Gertler, J. Fault Detection and Diagnosis in Engineering Systems. New York: Marcel Dekker, 1998. Accessed 2 August 2026.

[31] NASA Goddard Space Flight Center. GSFC-HDBK-8004: Guideline for Failure Modes and Effects Analysis and Risk Assessment. 2024. Accessed 2 August 2026.

[32] Bainbridge, L. "Ironies of Automation." Automatica 19, no. 6 (1983): 775–779. Accessed 2 August 2026.

[33] Parasuraman, R., and V. Riley. "Humans and Automation: Use, Misuse, Disuse, Abuse." Human Factors 39, no. 2 (1997): 230–253. Accessed 2 August 2026.

[34] National Institute of Standards and Technology. Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST AI 100-1, 2023. Accessed 2 August 2026.

[35] Higher-Order Pattern Narratives (Analysing Diagnostic Analysis). DumpAnalysis.org, August 2026. Accessed 2 August 2026.

[36] Schwartz, R., et al. Towards a Standard for Identifying and Managing Bias in Artificial Intelligence. NIST Special Publication 1270, 2022. Accessed 2 August 2026.

[37] Kelsey, J., J. Callas, and A. Clemm. RFC 5848: Signed Syslog Messages. Internet Engineering Task Force, May 2010. Accessed 2 August 2026.

[38] Biggio, B., B. Nelson, and P. Laskov. "Poisoning Attacks against Support Vector Machines." Proceedings of the 29th International Conference on Machine Learning, 2012. Accessed 2 August 2026.

[39] Unified Computer Diagnostics: Incorporating Hardware Narratology. DumpAnalysis.org, August 2026. Accessed 2 August 2026.

[40] OpenTelemetry. "Sampling." OpenTelemetry Documentation, updated 16 October 2025. Accessed 2 August 2026.

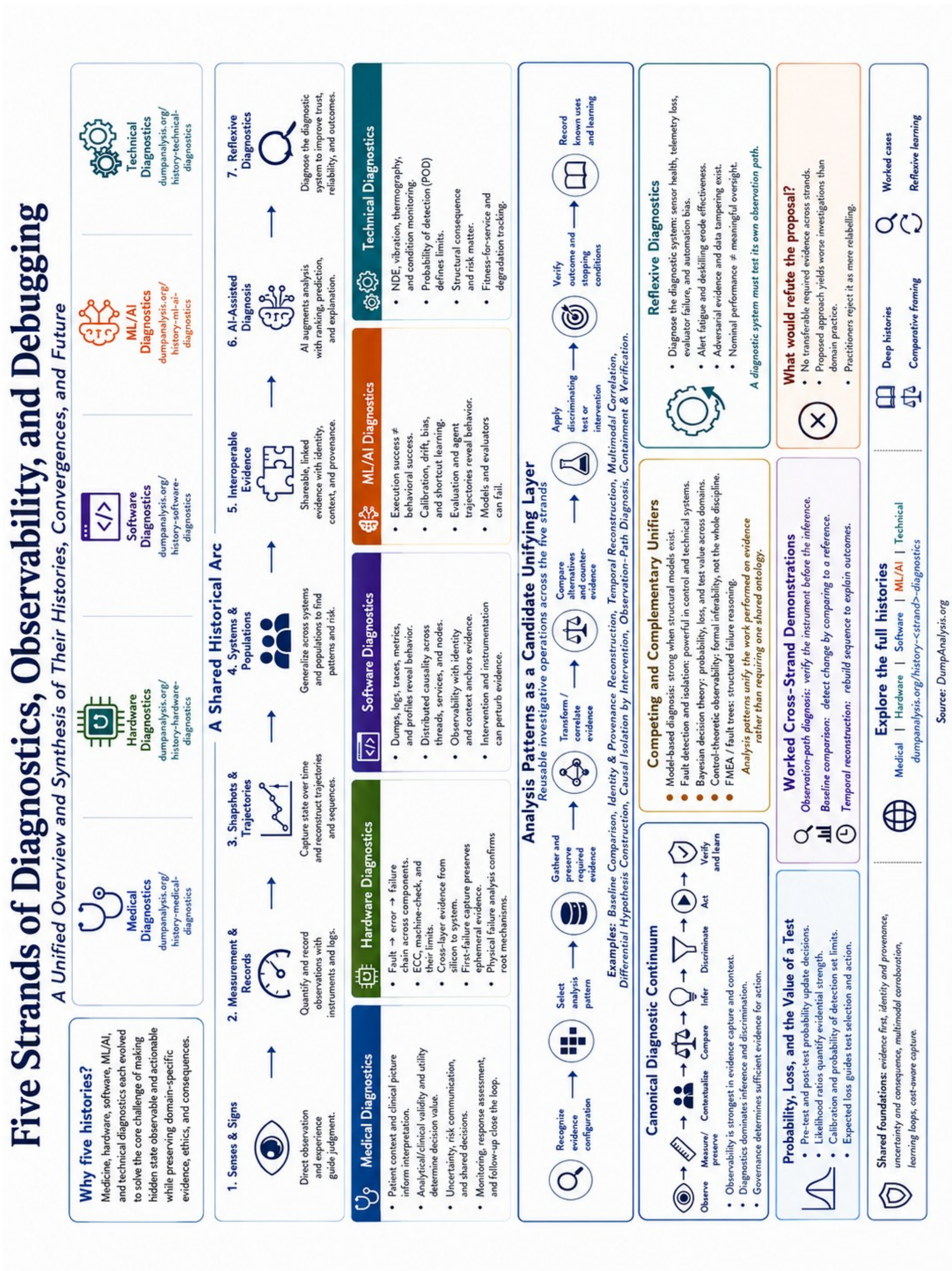

Figure 1. Five strands of diagnostics, observability, and debugging: historical arc, pattern-oriented bridge, reflexive layer, and diagnostic continuum.