
FRAMEWORK · WHITE PAPER

PDCA+

Version 2.0 · Public Review

A structural framework for how management-system disciplines align through a shared conjunction.

Author

Joacim Brandell

Companion to ISO 31000 · NIST 800-128 · ISO/IEC 27035 · ITIL 4 · ISO/IEC 27001

Executive Summary

Every management-system standard an organisation runs — ISO 31000 for risk, NIST 800-128 for configuration, ISO/IEC 27035 for incidents, ITIL 4 for change and events, ISO/IEC 27001 for information security — describes its own Plan–Do–Check–Act cycle. Each standard is internally coherent. None of them specifies how its cycle relates to the others.

In practice, this means your organisation runs several siloed cycles, and the alignment between them is reconstructed by hand — typically by a GRC team holding the whole picture in their heads. A configuration drift discovered by event management is a finding that risk management needs, but no standard tells you how to route it. A change approved by change management operates against a baseline that configuration management owns, but the standards leave the linkage informal. An incident closed by incident management produces lessons that risk and configuration both need to consume, but the lessons-learnt loop is built ad hoc.

The cost is real and recurring: duplicated work at every cycle position, gaps at the seams between management systems, alignment that lives in people's heads rather than in the framework, and a learning loop that closes within each silo but not across them.

THE STRUCTURAL CLAIM, IN ONE SENTENCE

The organisational lifecycle is not a structure that exists prior to its management systems and that they each apply. It is the structure that emerges when management-system disciplines align well with each other through a shared substrate.

PDCA+ describes that substrate, names the activities that become visible as strands of the lifecycle when alignment is good, and gives organisations a way to support the alignment deliberately rather than reconstruct it by hand.

The proposition

PDCA+ treats the organisational lifecycle as the byproduct of well-joined disciplines, held together by a strand conjunction.

A *discipline* is a management-system practice — risk management, configuration management, incident management, change management, and so on. Each operates on its own subject matter. A discipline is *bound* to PDCA+ when it has declared connection to the conjunction and participates in the joining work the conjunction supports.

A *strand* is an activity that becomes visible across well-joined disciplines because they all do it: context establishment, identification, analysis, treatment, acceptance, communication, review, monitoring. These activities are not pre-existing structural features that disciplines apply; they are the activities that show up everywhere when joining is done well, and they show up as strands of a shared lifecycle because the conjunction makes them legible across disciplines. The risk management standard articulates them most completely, because risk management's subject matter required the most thorough joining work — but the strands are not risk management's, and they are not any single standard's. They are what aligned disciplines look like together.

The *strand conjunction* is the shared substrate that makes good joining possible. It holds the schemas that let disciplines' inputs and outputs relate coherently, the taxonomies that classify what flows between them, the translation rules that make one standard's vocabulary legible to another, the measurement constructs that let each strand observe against shared definitions of adequacy (ISO 27004, NIST SP 800-55), and the cross-cutting capabilities — control management most prominently — that no single discipline owns but every discipline depends on. The conjunction provides these as services to disciplines that have declared connection. It holds no authority over them; it makes their alignment possible without governing their substance.

The “+” carries two meanings:

- **Structural** — the lifecycle is a continuously-running set of mutually-dependent strands held in a conjunction, not a four-phase iteration. The conjunction is where coordination, translation, instrumentation, and learning live as structural properties of the framework rather than as work distributed across management systems. PDCA's apparent four-phase shape is a description of one well-aligned discipline's operation; the actual structure across disciplines is the conjunction holding the strands as their byproduct.
- **Compositional** — disciplines participate in a shared lifecycle by declaring connection to the conjunction, rather than each running a private cycle. The organisation maintains one coordinating substrate while allowing multiple specialised disciplines to operate within it.

The lifecycle is continuous throughout, with one precise exception: when a discipline is drafting input to send into the conjunction, that input is momentarily off-cycle. As soon as it enters the conjunction, it participates in the joining. There are no closure events, no iteration boundaries, no “next cycle” — there is only continuous operation, interrupted at the edges by the moments when input is being prepared for entry.

The “+” also carries a quieter meaning: it is a negation of how PDCA has come to be applied. ISO standards have, over time, treated PDCA as a per-discipline property — each standard articulating its own four-phase cycle as if the activities belonged to it. PDCA+ rejects that

reading. The activities don't belong to the standards; they belong to the joining. PDCA+ is what PDCA looks like when the per-discipline framing is set aside.

What it gives you

	Siloed cycles	PDCA+
Activities	Each management system has its own activities (context establishment, identification, treatment, etc.)	Activities are strands that emerge across well-joined disciplines, supported by shared substrate
Translation between systems	Reconstructed by hand by GRC	Conjunction holds the schemas, taxonomies, translation rules, and shared capabilities that make alignment structural
Cross-cutting capabilities (measurement, control management)	Re-implemented per management system	Conjunction-housed, available to every bound discipline
Learning loop	Closes within each silo at periodic review events	Continuous: inadequacies in joining surface at the conjunction and are addressed by whichever disciplines are positioned to address them
Documentation	One manual per management system, each with its own activities	One conjunction description (substrate, schemas, services), plus one binding declaration per discipline
Governance posture	Each system owns its activities and accountability	Conjunction provides services to disciplines that declare connection; accountability remains with the disciplines

PDCA+ does not replace ISO 31000, NIST 800-128, ISO/IEC 27035, ITIL 4, or any other standard. Each remains the authority for its own subject matter. PDCA+ is the structural framework within which they align.

Practically, adoption means three things:

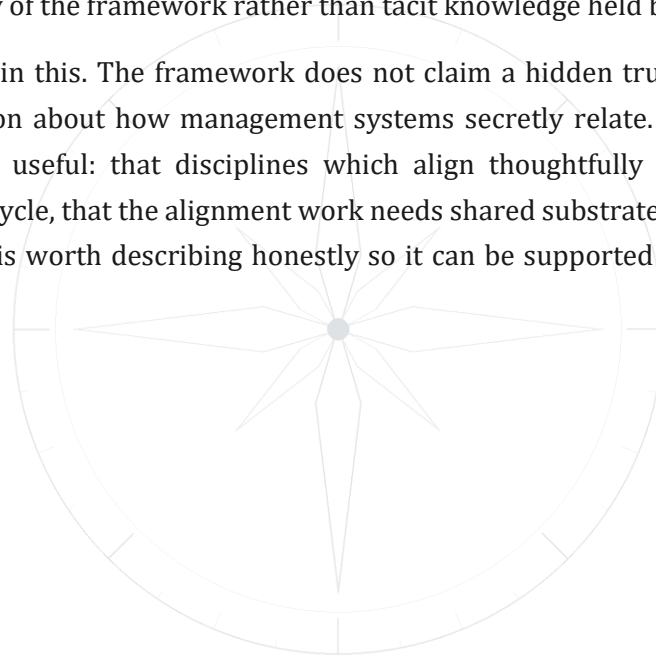
- **One conjunction description**, owned at the framework level, articulating what the conjunction holds (schemas, taxonomies, translation rules, measurement constructs,

control management as a shared capability) and what services it provides to bound disciplines.

- **One binding declaration per discipline**, naming which strands the discipline contributes to, which conjunction-held capabilities it consumes, and how its inputs and outputs flow through the conjunction's schemas. Disciplines bind by declaring connection; binding is opt-in.
- **One strand-conjunction discipline**, holding the substrate, surfacing inadequacies in joining, and operating “for those who declare connection” — providing services without exerting authority over the disciplines that consume them.

The standards stay where they are. The management systems keep their substance. What changes is that the activities those standards describe are recognised as belonging to the joining, not to the standards — and the alignment between disciplines becomes an explicit structural property of the framework rather than tacit knowledge held by individuals.

There is no magic in this. The framework does not claim a hidden truth about cycles or a structural revelation about how management systems secretly relate. It claims something simpler and more useful: that disciplines which align thoughtfully produce a coherent organisational lifecycle, that the alignment work needs shared substrate to be done well, and that the substrate is worth describing honestly so it can be supported deliberately. That is what PDCA+ is.



Conceptual Orientation

The structure PDCA+ describes

PDCA+ describes the organisational lifecycle as the byproduct of disciplines aligning through a shared conjunction. The framework can be understood by viewing four structural elements together:

Disciplines — The management-system practices operating on organisational subject matter — risk management, configuration management, incident management, change management, event management, and so on.

Strands — The activities that become visible across well-joined disciplines — context establishment, identification, analysis, evaluation, treatment, acceptance, review-and-communication.

The conjunction — The shared substrate that makes alignment possible: schemas, taxonomies, translation rules, measurement constructs, and cross-cutting capabilities such as control management.

Flows — The inputs, outputs, findings, classifications, observations, and acceptances moving continuously between disciplines through the conjunction.

The relationship between these elements is structural rather than sequential.

A discipline does not “run a cycle” independently. It participates continuously in strands that become coherent across the organisation because the conjunction makes them legible to other bound disciplines.

The conjunction does not govern the disciplines. It provides the substrate that allows them to align without reconstructing integration by hand.

1 The Problem PDCA+ Addresses

Classical PDCA, as inherited from Shewhart and Deming, is a description of how an aligned practice operates: plan what you intend, do it, check what happened, act on what you found, and continue. As inherited, it is not a structural framework — it is a description of the shape that thoughtful, iterative work takes.

When ISO standards adopted PDCA as the structuring principle for management systems, something happened to it. Each standard articulated its own PDCA cycle: ISO 9001's quality cycle, ISO/IEC 27001's information-security cycle, ISO 14001's environmental cycle, ISO 45001's occupational-health cycle, ISO 31000's risk cycle, ISO/IEC 27035's incident cycle. The activities each cycle named — context establishment, identification, treatment, acceptance, communication, review — were articulated as that standard's activities, owned by the management system that standard governed.

This per-standard articulation is internally coherent within each standard. It is also the source of the problem PDCA+ addresses.

In practice, an organisation's risk management, configuration management, change management, incident management, and event management are deeply entangled. A configuration drift discovered by event management is a finding that risk management needs. A change approved by change management is operating against a baseline that configuration management owns. An incident closed by incident management produces lessons that risk and configuration both need to consume. The standards that govern each of these (ISO 31000, NIST 800-128, ISO/IEC 27035, ITIL 4) each describe their own PDCA-shaped cycle, but the standards do not — and cannot — specify how the cycles relate to each other.

The result, in most organisations, is a set of siloed cycles. Each management system runs its own context establishment, its own identification, its own treatment, its own acceptance, its own communication and review. The activities are the same activities, applied to different subject matter, but each system implements them independently. The relationships between them are reconstructed by hand at integration time, usually by a GRC team holding the whole picture in their heads.

The siloed reading is what PDCA+ negates. It is not that the standards are wrong about the activities; they describe the activities accurately. It is that the activities they describe do not belong to them.

The reframe

PDCA+ proposes that the activities each standard describes — context establishment, identification, analysis, treatment, acceptance, communication, review, monitoring — are not

properties of any management system. They are what well-joined disciplines all do. They become visible as strands of a shared organisational lifecycle when disciplines align with each other through a conjunction that holds the schemas, taxonomies, translation rules, and cross-cutting capabilities the alignment needs.

Risk management's standard happens to articulate the activities most completely, because risk management's subject matter required the most thorough joining work. ISO 31000's vocabulary is therefore a useful reference for naming the strands. But the strands are not risk management's. Configuration management does context establishment — it establishes the configuration baseline against which deviation is named. Incident management does context establishment — it establishes the incident-handling context within which incidents are recognised and responded to. Change management does it; ITIL 4 practices do it; every discipline that operates thoughtfully on subject matter does it. The standards don't all name it equally clearly, but the activity is the same one.

Once this reframe is in place, several things change.

The cycle is no longer a structure each discipline runs in private. It is the structure that emerges across disciplines when their joining is supported. There is no “the risk management cycle” and “the configuration management cycle” running in parallel; there is one organisational lifecycle, and risk management and configuration management are two disciplines whose work participates in it.

Pre-existing PDCA's four-phase shape is not the structure of the lifecycle. It is the shape that becomes visible when you look at one discipline's complete application of the strands. When multiple disciplines align, the four-phase shape gives way to something more like a continuously-running set of mutually-dependent strands held together by the conjunction.

The conjunction is where the work that was previously distributed — vocabulary translation, schema alignment, classification taxonomies, cross-cutting measurement, control management — lives as a structural property of the framework. The conjunction is not a governance layer above the disciplines. It is a substrate beneath them. It provides services; it does not exert authority. Disciplines participate by declaring connection.

In what follows, §2 describes the conjunction first — what it holds, what services it provides — because the strands and the disciplines both depend on it. §3 describes the strands as activities that become visible across well-joined disciplines. §4 describes how disciplines bind to the framework. §5 traces a single finding through the structure as a worked example. §6 relates PDCA+ to existing standards, including how it sits alongside Annex SL.

There is no claim of structural revelation here. The framework describes what well-aligned disciplines produce when they align, and what substrate they need to align well. It earns its

abstraction by making that alignment supportable in the framework rather than reconstructable by hand each time. That is all it does, and it is enough.



2 The Strand Conjunction

The strand conjunction is the substrate that makes alignment between disciplines possible. It is not above the disciplines and not below them. It sits beside them, holding what they need in common.

What it holds is, at root, one kind of thing: the artefacts and tooling that disciplines depend on but that no single discipline can own without distorting it. A taxonomy that classifies findings is no use if only one discipline uses it; it has to be common ground. A schema for how a configuration record relates to a risk register entry is no use if only configuration management or only risk management understands it; it has to be legible to both. A measurement construct against which control effectiveness is observed is no use if it lives inside one management system; it has to be shared.

In practice, what the conjunction holds falls into four overlapping categories. They are not separate things the conjunction does; they are different views of the same substrate.

Schemas. The structural definitions of what flows between disciplines. A configuration record has a shape; a risk register entry has a shape; an incident report has a shape; a control assessment has a shape. The schemas the conjunction holds are the ones that let outputs from one discipline be consumed as inputs by another without each discipline having to translate the other's idiom. Where two disciplines' standards specify their own artefact shapes (NIST RMF's System Security Plan and ISO 27001's Statement of Applicability, for example), the conjunction holds the schema-level mapping that lets them share evidence without loss.

Taxonomies. The classification systems by which findings, events, risks, changes, and other objects are categorised so they can be routed and acted on coherently. The classification taxonomy that decides whether a deviation is a “configuration drift” or a “change-authorisation failure” is conjunction-held, because it determines how findings flow between disciplines. A taxonomy maintained inside a single discipline cannot do this work; it can only describe how that discipline sees things.

Translation rules. The vocabulary mappings between standards. When ISO 31000 says “risk owner” and NIST RMF says “Authorizing Official,” the two are not synonyms — they carry different institutional authority — but the artefacts each role produces have specifiable relationships. The conjunction holds the rules that say which artefact under one regime corresponds to which artefact under another, with what scope conditions and what caveats. Translation here is artefactual, not institutional. Roles are recorded separately, in the disciplines' own accountability documentation; the conjunction does not translate authority.

Cross-cutting capabilities. The disciplines that no single management system owns but every management system depends on. Control management is the most prominent example:

ISO 27001, NIST RMF, ISO 31000, and configuration management all rely on a coherent picture of what controls exist, how they are specified, how their effectiveness is observed, and how they relate to the risks and obligations they address. If each management system maintained its own control inventory, the inventories would diverge, and the divergence would be reconciled by hand. The conjunction holds control management as a shared capability that all bound disciplines consume from. Measurement, in the ISO 27004 / NIST SP 800-55 sense, is another such capability: the constructs against which strands observe adequacy are not individual disciplines' constructs, they are conjunction-held.

These four overlap. A schema describing how a control assessment record relates to a risk register entry is also a translation rule between two standards' vocabularies and is also part of the cross-cutting control-management capability. The conjunction does not maintain four separate registers; it maintains one substrate that can be viewed from several angles depending on what a discipline is trying to do at a given moment.

What the conjunction does

The conjunction does three things, all of them continuous, none of them periodic.

It provides. Bound disciplines consume the conjunction's services as they operate. When configuration management needs to relate a baseline deviation to the risk register, it consumes the schema that defines that relationship and the taxonomy that classifies the deviation. When risk management needs to assess control effectiveness, it consumes the measurement constructs the conjunction holds. The conjunction is always available; disciplines reach into it as their work requires.

It observes. The conjunction continuously observes whether what flows through it is coherent. When two disciplines' inputs to a shared schema produce inconsistencies, the conjunction sees them. When a taxonomy is being asked to classify findings it doesn't fit, the conjunction sees it. When a translation rule produces artefacts that don't carry the meaning they were supposed to carry, the conjunction sees that too. Observation here is not an audit activity performed periodically against the disciplines; it is a continuous property of the substrate. The conjunction sees what passes through it because that's what holding a substrate means.

It surfaces. When the conjunction's observations identify inadequacies — places where alignment is failing because the substrate isn't supporting it well — it surfaces those findings to the disciplines positioned to address them. The conjunction does not fix the inadequacy; it makes it visible. A taxonomy that consistently misclassifies a category of findings is the conjunction's problem to surface and the relevant disciplines' problem to revise. A schema that doesn't accommodate an emerging artefact type is similarly surfaced. The conjunction's

authority here is observational, not corrective. Findings flow to whoever has the authority to act on them, which is always one or more bound disciplines.

These three are not phases of a meta-cycle. There is no cycle at the meta level; there is continuous joining work, with the conjunction continuously providing, observing, and surfacing. The earlier versions of this framework described a “meta-loop” closing inadequacy findings back to a “next pre-binding event.” That description was inherited from PDCA's iteration vocabulary and does not fit what actually happens. Inadequacies are surfaced when they become visible; they are addressed when disciplines take them up; nothing closes because nothing was an iteration to begin with.

The governance posture

The conjunction operates “for those who declare connection.” This is a structural posture, not a political compromise.

A discipline that has declared connection to the conjunction binds to its services and is observable through them. Its inputs flow through conjunction-held schemas; its outputs become legible to other bound disciplines through conjunction-held taxonomies and translation rules; its work participates in the cross-cutting capabilities the conjunction maintains. The discipline retains full authority over its substantive decisions — what risks it accepts, what configurations it authorises, what incidents it declares closed. The conjunction does not govern those decisions; it makes them legible to other disciplines and surfaces inadequacies in how the legibility is working.

A discipline that has not declared connection still operates. It just doesn't participate in the joining. Its activities don't become strands of the shared lifecycle, because there's no substrate making them legible across the boundary. This is not a punishment; it is what non-participation structurally is. Many disciplines in many organisations operate without conjunction support, and the framework has nothing to say about whether that is wrong — only about what is gained when they do connect.

The conjunction therefore holds no authority over disciplines. It cannot compel a discipline to bind, cannot require a bound discipline to revise its practices, cannot override a discipline's substantive decisions. What it can do is provide services well enough that binding is the obviously useful thing for a discipline to do, and surface inadequacies clearly enough that bound disciplines have the information they need to revise themselves.

This posture is what allows the framework to be adopted incrementally. An organisation does not need to bind all its management systems at once. It binds whichever systems are ready to declare connection, and the conjunction supports the alignment between them. As more systems bind, more strands become visible across the lifecycle, and the substrate the

conjunction holds expands to support them. Adoption is additive; non-binding is structurally accommodated.

The conjunction's own structure

Mature implementations of the conjunction develop their own internal organisation. In practice, this often looks like a continuous instrumentation discipline — identifying what needs to be measured, designing the measures, implementing them, collecting data, analysing it, and reporting findings — but the specifics depend on the organisation's scale and the disciplines it has bound. ISO/IEC 27004 describes one such organisation; NIST SP 800-55 describes another; either can be used as a reference when implementing the conjunction in practice.

What is structural, regardless of internal organisation, is that the conjunction is one discipline. It does not split into “the schema discipline” and “the taxonomy discipline” and “the measurement discipline” any more than it splits its three functions into separate teams. The substrate is one substrate; the discipline that holds it is one discipline. Internal specialisation may exist for capacity reasons, but the conjunction's coherence as a single substrate is what lets it do its work.

In standards-vocabulary terms, the conjunction occupies the territory that ISO 9001 calls “management review,” that ISO/IEC 27001 calls “performance evaluation,” and that NIST SP 800-53 calls “control assessment” — but elevated from periodic review activities to continuous substrate. It also includes the work that ISO/IEC 27004 calls “measurement” and that NIST SP 800-55 calls “performance measurement.” The conjunction is not new work; it is the recognition that work already being done in scattered places across the standards belongs together as one substrate, and the description of what it looks like when it is held together coherently.

3 The Strands

When disciplines align well through the conjunction, certain activities become visible across all of them. These activities are the strands of the shared lifecycle. They are not steps any discipline runs in sequence; they are continuous activities that bound disciplines all do, made legible to each other by the conjunction's substrate.

The strands are named with vocabulary inherited from the risk management standard. ISO 31000 articulates them most completely because risk management's subject matter — uncertainty across the whole organisation — required the most thorough joining work to handle coherently. The vocabulary is therefore useful as reference. The strands themselves are not risk management's, and the names are not claims about ownership. They are convenient labels for activities that show up across well-joined disciplines regardless of subject matter.

Seven strands are typically visible. Other strands may emerge in specific organisational contexts; what follows is the common set that appears wherever multiple management-system disciplines align through a shared conjunction.

Context establishment. The activity by which a discipline determines what it is operating on, within what scope, against what criteria, with what dependencies. Risk management establishes risk context; configuration management establishes the baseline against which deviation is named; incident management establishes the context within which an event becomes an incident; change management establishes the context that defines whether a change is in scope. Each discipline does this for its own subject matter, and the conjunction holds the schemas that let each discipline's context be related to the others' — so that, for instance, the configuration baseline informs the risk context and vice versa.

Identification. The activity by which what is faced becomes named. Risk identification names risks; configuration identification names the actual state against the baseline; incident identification names what has occurred and what it concerns; change identification names what is being changed. Identification is continuous in well-joined disciplines, not episodic — the discipline is always in the process of recognising what is currently faced, with the conjunction's taxonomies providing the classification structure that makes the identification legible to other bound disciplines.

Analysis. The activity by which what has been identified is examined for its properties — likelihood, impact, scope, character, dependencies. Risk analysis examines risks; configuration analysis examines deviations; incident analysis examines incidents; change analysis examines changes. The measurement constructs the conjunction holds (ISO 27004,

NIST SP 800-55) provide the shared definitions of adequacy, severity, effectiveness, and similar properties against which analysis observes.

Evaluation. The activity by which the results of analysis are weighed against criteria established in context. Whether to treat a risk, whether to authorise a deviation, whether to escalate an incident, whether to approve a change — all are evaluation work. Evaluation is the strand at which judgment is exercised most explicitly, and it is also where authority structures are most visible: the discipline's accountability documentation specifies who exercises the judgment, against what delegation, with what record-keeping.

Treatment. The activity by which the discipline acts on what evaluation has decided. Risk treatment commits responses to risks; configuration treatment commits responses to deviations (revert, document, escalate); incident treatment commits responses to incidents; change treatment commits the change itself. Treatment is the strand most often described in standards as “the substantive work” of the discipline, and it is — but it is no more substantive than the other strands. It is simply the strand at which the discipline most visibly interacts with the world it operates on.

Acceptance. The activity by which residual posture is bound. After treatment, what remains? Risk acceptance binds the residual risk the organisation will operate with; configuration acceptance binds the operational posture of the configured system; incident acceptance closes the incident with its lessons learned; change acceptance recognises the change as in effect. Acceptance is conditional in well-joined disciplines — conditional on the substrate continuing to support the bases on which acceptance was given. When the conjunction surfaces inadequacies in those bases, acceptance is reopened in the discipline that owns it.

Review and Communication. The activity by which the discipline observes itself and the world it operates on, and by which what is observed becomes available to those who need it. These are two surfaces of one strand, not two strands. Review is what the activity looks like from the inside — continuous observation against criteria established in context, testing whether commitments are still supported by the evidence they were made on. Communication is what the same activity looks like from the outside — the observation reaching the people and disciplines who need it, both within the organisation (stakeholders, accountable parties) and across bound disciplines (whose work depends on the information).

ISO 31000 articulates these as two parallel clauses — “communication and consultation” alongside “monitoring and review” — and other standards follow similar splits. The split is bibliographic, not structural. In well-joined disciplines the observation and its carrying are one continuous activity: nothing is observed that isn't communicated to those who need it, and nothing is communicated that wasn't observed. Treating them as separate processes is what produces the duplication that PDCA+ negates — each management system implementing its own communication function and its own review function, when the

conjunction's substrate already carries most of the work and what remains is one strand the discipline applies to its own subject matter.

The conjunction's role here is the same as for any strand: it holds the schemas and taxonomies that make the discipline's observations legible to other bound disciplines without translation effort, and it provides the measurement constructs (ISO 27004, NIST SP 800-55) against which observation is calibrated. What remains for the discipline to do is the part the substrate cannot do for it — the observation that requires the discipline's own judgment, and the communication that requires explanatory context the schemas cannot carry.

How the strands relate

The strands are mutually dependent, not sequential. Identification depends on context; analysis depends on identification; evaluation depends on analysis; treatment depends on evaluation; acceptance depends on treatment. So far, this looks like a sequence. But review and monitoring observe across all of them and continuously feed back into identification when conditions warrant. Review and Communication runs across all the others, observing them against criteria and carrying findings to where they are needed. Context is continuously revisited as the discipline learns about its own subject matter. The strands operate concurrently, with dependencies between them but without a fixed order in which they must be encountered.

This is what distinguishes the strand picture from the four-phase PDCA picture. PDCA's four phases imply sequence — Plan first, then Do, then Check, then Act, then back to Plan. The sequence is a useful approximation when describing one well-aligned discipline's complete operation; it captures what most often happens in what order. But it is not the structure. The structure is mutual dependence between continuously-running activities, with the conjunction's substrate making the dependencies tractable.

The single discontinuity worth naming, which §1 anticipated: when a discipline is drafting input to the conjunction, that input is momentarily off-cycle. A risk register entry being composed, a configuration record being prepared, an incident report being written — all are off-cycle until they enter the conjunction. As soon as they enter, they participate in the joining, and the strand they enter at (identification, analysis, treatment, etc.) carries them. The discontinuity is at the edge of input, not within the cycle's operation.

Why the strands look like phases when only one discipline is observed

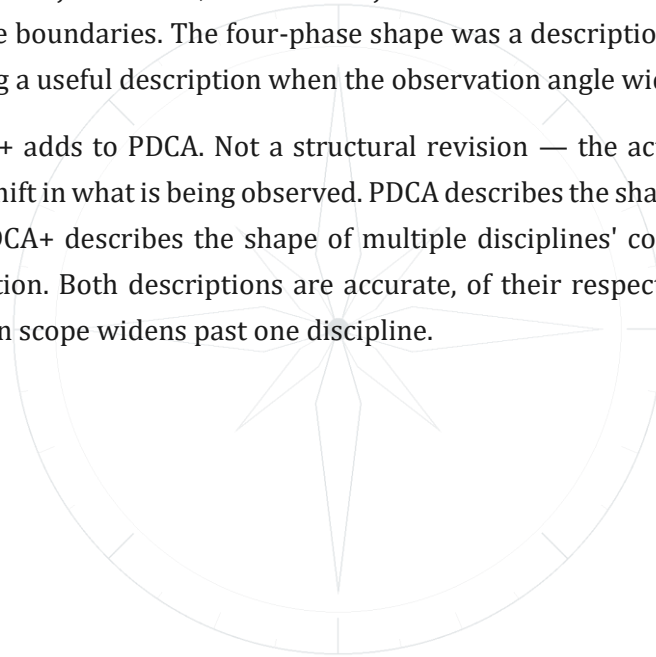
A reader familiar with PDCA-shaped standards may notice that the strands described here line up suspiciously well with the four phases of the classical cycle. Context establishment looks like Plan; identification, analysis, evaluation, and treatment look like Do; review and monitoring look like Check; acceptance and the corrective trigger from review look like Act.

This alignment is not accidental, and recognising why it occurs is the cleanest way to see what the strand picture actually changes.

When only one discipline is observed, the strands appear sequential because that discipline applies them in roughly the order the dependencies imply. Context first, then identification within the context, then analysis of what was identified, and so on. From a single discipline's vantage point, the cycle appears to have four (or five, or six, depending on how the standard breaks them out) phases, and the standard articulating that discipline describes them as such.

When multiple disciplines are observed together, the apparent sequence dissolves. Configuration management's identification is happening continuously while risk management's analysis is happening continuously while incident management's communication is happening continuously. Each discipline is at every strand simultaneously, applied to its own subject matter, and the conjunction's substrate is making their work coherent across the boundaries. The four-phase shape was a description of one observation angle; it stops being a useful description when the observation angle widens.

This is what PDCA+ adds to PDCA. Not a structural revision — the activities are the same activities — but a shift in what is being observed. PDCA describes the shape of one discipline's complete work. PDCA+ describes the shape of multiple disciplines' complete work joining through a conjunction. Both descriptions are accurate, of their respective scope. PDCA+ is what happens when scope widens past one discipline.



4 How Disciplines Bind to PDCA+

A management system is the organisational practice — risk management, configuration management, incident management, change management, and so on. A *binding* is what makes that practice a participant in the shared lifecycle: the management system declares connection to the conjunction, names which strands its work contributes to, and consumes the conjunction's services as its work requires.

Binding is opt-in and structurally additive. A management system that has not declared connection still operates; it just doesn't participate in the joining, and its activities don't become strands of the shared lifecycle as far as the framework is concerned. As more systems bind, more of the organisation's work participates in the shared substrate, and the conjunction's services correspondingly expand to support the new participants.

What a binding declares

A binding declaration names four things.

Which strands the discipline contributes to. Not every discipline contributes to every strand. A configuration management discipline following NIST 800-128 contributes to identification (deviations against baseline), analysis, treatment, and acceptance — and to review-and-communication, through the substrate. It may not contribute to context establishment if it inherits its baseline from upstream rather than establishing it. A partial-binding discipline like event management contributes only to identification (recognising deviations) and dispatches downstream by classification, contributing nothing further to treatment or acceptance because that work belongs to other bound disciplines. The declaration names what the discipline brings.

Which conjunction-held services it consumes. Every bound discipline consumes some subset of what the conjunction holds — schemas for the artefacts it produces and consumes, taxonomies for the classifications it makes, translation rules for the standards-vocabularies it interfaces with, measurement constructs for the observations it performs, and shared capabilities (control management most prominently) for the cross-cutting work no single discipline owns. The declaration names which services the discipline relies on, so that changes to those services can be surfaced to it.

On which dimensional axis the discipline operates. A discipline operates on a particular decomposition of the organisation. ISO 27001 operates on ISMS scope. NIST RMF operates on system boundary and impact tier. Configuration management operates on configuration baselines. Incident management operates on the incident-handling lifecycle. The axis matters because it determines what the discipline's outputs are *about* — a risk acceptance under ISO

27001's ISMS-scope axis is not the same artefact as an authorisation decision under NIST RMF's boundary-tier axis, even when both concern the same systems. The conjunction's translation rules accommodate the difference; the binding declaration names the axis so the conjunction knows which translations to apply.

How the discipline's inputs and outputs flow. When the discipline produces an output that other bound disciplines may consume, the binding declares the schema the output conforms to and the taxonomy under which it is classified. When the discipline consumes inputs from other disciplines, it declares the same in reverse. This is what makes routing between disciplines structural rather than coordination-by-hand: a finding produced by event management against a known schema, classified under a known taxonomy, with a known axis, can be routed to whichever bound discipline is positioned to receive it without any human deciding where it goes.

Patterns of binding

Disciplines bind in different shapes depending on their subject matter and the standards that articulate them. Three patterns recur often enough to be worth naming, though they are not exhaustive — many bindings combine elements of several.

Full binding. The discipline contributes to most or all strands in its own right, applying them to its subject matter without inheriting from another discipline. ISO 31000-articulated risk management is the canonical example: it establishes its own context, identifies and analyses and evaluates and treats and accepts within that context, and applies review-and-communication continuously throughout. Full bindings are the most demanding to specify but also the most self-contained.

Inherited binding. The discipline does not contribute to all strands in its own right; it inherits from an upstream discipline's contributions. A change management discipline typically inherits its context (the configuration baseline against which changes are evaluated) from configuration management's context establishment, rather than establishing its own. An incident management discipline typically inherits both context and identification from upstream event management. Inherited bindings reduce duplication: the upstream discipline's work is consumed rather than redone.

Partial binding. The discipline contributes only to some strands and routes the others to other bound disciplines. Event management is the canonical example: it contributes to identification (recognising what has occurred) and dispatches downstream by classification. It does not contribute to treatment or acceptance, because those are the responsibility of the disciplines event management routes to. Partial bindings are how the framework accommodates disciplines whose substantive role is recognition-and-routing rather than substantive response.

Multi-strand operator steps. Some disciplines, by the standards that articulate them, perform multiple strands' work in a single operator step. NIST 800-128's articulation of configuration management is the most prominent example: its monitoring activity simultaneously identifies deviations, evaluates them against the baseline, treats them (revert / document / escalate), and accepts the resulting posture. The strands are still distinguishable in what the activity produces — there is identification, there is evaluation, there is treatment, there is acceptance — but they are bound in one step rather than four. The binding declaration names which strands the step covers, so that the conjunction can route inputs and outputs correctly even though the discipline's own articulation does not separate the strands.

What the binding does not declare

Three things are deliberately outside the binding declaration's scope.

Authority and accountability remain with the discipline. The binding does not relocate authority over substantive decisions — what risks to accept, what configurations to authorise, what incidents to declare closed. Those authorities are recorded in the discipline's own accountability documentation (often a role authority matrix maintained alongside the framework, particularly for organisations operating across regimes like ISO 27001 and NIST RMF). The conjunction translates the artefacts those authorities produce, not the authorities themselves. A “risk owner” under ISO 31000 and an “Authorizing Official” under NIST RMF are not equivalent roles; the matrix records how named individuals carry both kinds of authority for the systems and scopes that overlap.

Internal organisation of the discipline is not specified. How a bound discipline organises itself internally — which teams do which work, what tooling it uses, how its activities are scheduled — is its own concern. The binding declares the discipline's interface to the framework, not its internals. Two organisations binding the same discipline may organise it very differently; the binding describes only what makes the discipline a participant in the shared lifecycle.

The discipline's substantive standards remain authoritative. The binding does not displace ISO 31000, NIST 800-128, ISO/IEC 27035, or any other standard the discipline implements. Those standards remain the authority for the discipline's subject matter. The binding describes how the discipline's work, conducted under its own standards, participates in the shared lifecycle PDCA+ describes.

Adoption is incremental

Because binding is opt-in and structurally additive, adoption does not require the organisation to bind all its management systems at once. An organisation typically begins by binding two or three disciplines that have the strongest interdependencies — often risk

management with configuration management, or risk management with incident management — and expands as the conjunction's substrate proves its value. Disciplines that remain unbound continue to operate; they simply don't participate in the joining, and the framework has nothing to say about whether that's appropriate for them.

The conjunction's substrate also expands incrementally. An organisation does not need to specify all schemas, all taxonomies, all translation rules, all measurement constructs before binding its first discipline. The substrate is built as bindings require it. The first binding establishes the schemas and taxonomies its outputs need; the second adds what it requires; over time the substrate accumulates into something comprehensive. What matters is that the substrate is held coherently — one conjunction maintaining one substrate, even as it grows — rather than fragmented across disciplines.



5 Worked Example

What follows traces a single finding — an unexpected configuration change on a production server — as it moves through a PDCA+ structure. The point is not the operational detail. It is the structural one: notice how the finding is handled by bound disciplines drawing on the conjunction's substrate, with no person reconstructing the integration by hand.

5.1 Setting

A mid-sized financial services organisation has bound four disciplines to its PDCA+ structure: event management (partial binding), configuration management (multi-strand operator steps, articulated by NIST 800-128), risk management (full binding, articulated by ISO 31000), and incident management (inherited binding, articulated by ISO/IEC 27035). The conjunction is operated by the organisation's GRC function as a continuous discipline. ISO/IEC 27001 governs the ISMS scope under which these disciplines operate; NIST RMF governs the system-boundary-and-tier scope for the systems within that ISMS. The role authority matrix documents how named accountabilities carry across both regimes.

The configuration baseline for the affected system class was established eight months ago. The risk register's last formal review was four months ago. Both rest on context-establishment work done at the start of the current annual cycle and held in the conjunction's substrate as schemas and taxonomies the bound disciplines continuously consume.

5.2 What happens

1. Event management identifies a deviation. At 03:47, the event-monitoring tooling registers an unexpected change to a TLS configuration on a production payment-processing server. The change does not match any approved entry in the change register.

Event management is a partial-binding discipline. It contributes to the identification strand and dispatches downstream; it does not contribute to evaluation, treatment, or acceptance for findings of this kind. Its identification work names what has been observed: “unauthorised configuration change, payment-processing tier, TLS settings,” classified under the conjunction's classification taxonomy.

The dispatch is structural, not coordinated. The taxonomy says deviations of this type, on systems of this tier, route to configuration management for substantive response, with parallel notification to risk management. Event management's output enters the conjunction at the schema for configuration-deviation findings; the conjunction's taxonomy classifies it; the bound disciplines whose declarations name them as receivers for this classification

consume it. No person reads a runbook and decides where the finding goes. The substrate carries it.

2. Configuration management responds. Configuration management is a multi-strand-operator-steps discipline. Its monitoring activity, articulated by NIST 800-128, simultaneously contributes to identification (the deviation is recognised against the baseline), evaluation (the deviation is examined for its character and severity), treatment (a response is committed), and acceptance (the resulting operational posture is bound).

In this case, configuration management determines that the TLS change is not technically harmful in itself — the new settings are stricter than the baseline, not weaker — but that the baseline was changed without authorisation. The treatment is to document the change, hold the system in its current state pending baseline review, and flag the unauthorised-change pathway as the matter requiring attention. The acceptance binds the residual posture: the organisation operates, for now, with a system whose configuration differs from its authorised baseline, conditional on the baseline-review activity completing within a defined window.

Configuration management's outputs enter the conjunction's substrate: the documented deviation, the held posture, the flagged pathway. They become available to other bound disciplines through the schemas and taxonomies the conjunction holds.

3. The conjunction observes a coherence problem. The conjunction continuously observes whether what flows through it is coherent. Configuration management's just-bound posture satisfies the configuration-management criteria — the deviation is documented, the system is held, the pathway is flagged — but it visibly conflicts with a cross-cutting policy criterion that configuration management's substrate does not enforce: the policy commitment that all production configuration changes pass through change management's approval pathway.

The conjunction sees the conflict because both the configuration-management acceptance and the cross-cutting policy commitment are held in its substrate, made commensurable by its schemas. Neither configuration management nor change management could see the conflict alone — each works within its own bindings, against its own criteria. The conjunction sees it because it holds the substrate in which both criteria live.

4. The conjunction surfaces the finding to risk management. The conjunction does not resolve the conflict. It surfaces the finding to the disciplines positioned to address it. Risk management's binding declares it as a consumer of findings that affect the evidence base on which its acceptances rest. The current finding affects exactly that base: the risk register's entry for “unauthorised configuration changes in production” was rated low likelihood on the assumption that change-management controls were operating as designed, and the current finding is evidence that the assumption may not hold.

The finding reaches risk management at its identification strand. Risk management's review-and-communication strand has been continuously observing against the criteria its bindings established; the conjunction's surfacing of the finding is exactly the kind of input that strand consumes. The resulting work — possibly a new analysis, possibly a revised treatment, certainly a revised acceptance — proceeds within risk management's binding. The conjunction's role is complete once the finding is surfaced.

5. The conjunction observes a recurring pattern. Over the following weeks, the conjunction continues to observe what flows through it. The TLS finding is the third instance this quarter in which event management's classification routed a finding to configuration management as a “configuration deviation” when the underlying matter was more accurately characterised as a “change-authorisation failure.” Configuration management is structurally well-equipped for the former but not the right primary owner of the latter — it can only document, hold, and flag, leaving the substantive remediation of the authorisation pathway to a downstream hand-off that does not occur cleanly.

This is not a finding about any bound discipline's substantive work. Each discipline did what its binding required; their substantive work is fine. The finding is about the conjunction's own substrate — specifically about the classification taxonomy that decides where event management's outputs route. The taxonomy is treating two different kinds of deviation as one, and that treatment is now visibly inadequate to the work it has to do.

The conjunction surfaces this finding too. It does not surface it to any single discipline, because no single discipline owns the taxonomy. It surfaces it to those who maintain the substrate — the conjunction discipline itself, with input from the bound disciplines whose work the taxonomy affects.

6. The substrate is revised. At the next opportunity for taxonomy revision — in this organisation, a quarterly governance forum where substrate revisions are made by the conjunction discipline in consultation with bound disciplines — the taxonomy is revised. A new dispatch class is established for findings whose primary character is change-authorisation failure rather than configuration deviation. The routing rules are updated: such findings now route to a different downstream pattern (a partial-binding discipline that recognises the authorisation failure and dispatches to both change management and risk management, rather than to configuration management as primary).

The revision lives in the conjunction's substrate. From the moment it is in force, every bound discipline whose work passes through that taxonomy operates against the revised version. Event management, on its next finding of this type, classifies against the new taxonomy. Configuration management receives a different — and more appropriate — flow of findings.

Risk management receives change-authorisation evidence directly from a routing pattern designed to surface it, rather than indirectly through later observation of policy variance.

Nothing closed. No iteration completed. The substrate was revised because the conjunction surfaced an inadequacy, and the disciplines positioned to revise it took it up. The framework's own learning happens the same way the rest of its work does: continuously, through well-joined disciplines drawing on a coherent substrate.

5.3 What this example illustrates

Five things, each tied to a structural claim the framework makes.

Routing between disciplines is structural. Every hand-off in the example happened through the conjunction's substrate. Event management's dispatch to configuration management ran through the classification taxonomy. The conjunction's surfacing of the policy variance to risk management ran through the schemas that made both criteria commensurable. The recurring-pattern detection and its surfacing for substrate revision ran through the conjunction's continuous observation. No hand-off was implicit; none required a person holding the picture in their head.

The conjunction's three functions operated continuously. Provision was continuous (every bound discipline drew on the substrate as its work required). Observation was continuous (the conjunction saw what passed through it because that's what holding the substrate means). Surfacing was continuous (findings flowed to the disciplines positioned to address them as they became visible). No function fired periodically. No closure event occurred. The example traced one path through the framework's continuous operation, not a sequence of cycle-bounded iterations.

The substrate accumulated learning. The classification taxonomy that worked adequately at the start of the example was inadequate by its end. The substrate was revised to accommodate what had been learned, and the revision propagated automatically to every bound discipline operating through the taxonomy. The framework's learning is the substrate's revision; the substrate's revision is what the bound disciplines next consume.

No discipline was made to do anyone else's work. Event management did identification and dispatch. Configuration management did its multi-strand response. Risk management did its full-binding work on its own subject matter. Each discipline operated within its own binding, on its own subject matter, against its own standards. The substrate did the cross-discipline work — making outputs legible across boundaries, surfacing inadequacies that no single discipline could see alone, accumulating learning that no single discipline could sustain alone.

Authority remained where it belongs. Configuration management's acceptance of the held posture was configuration management's decision, made under its own authority. Risk management's eventual revised acceptance was risk management's decision, made under its own. The conjunction surfaced findings; it did not authorise responses. The role authority matrix recorded who had authority to make what decisions across the regimes the disciplines operated under, and the framework respected those records throughout.

What is *not* in the example is also instructive. No bound discipline reimplemented its own communication or review function. No human routed findings between disciplines. No GRC team reconstructed integration at the seams. The substrate did this work. That is what PDCA+ is for, and it is the simplest demonstration of why the framework earns its abstraction: it makes the integration that is currently distributed across people and reconstructed by hand into a structural property of how disciplines align.



6 Relation to Existing Standards

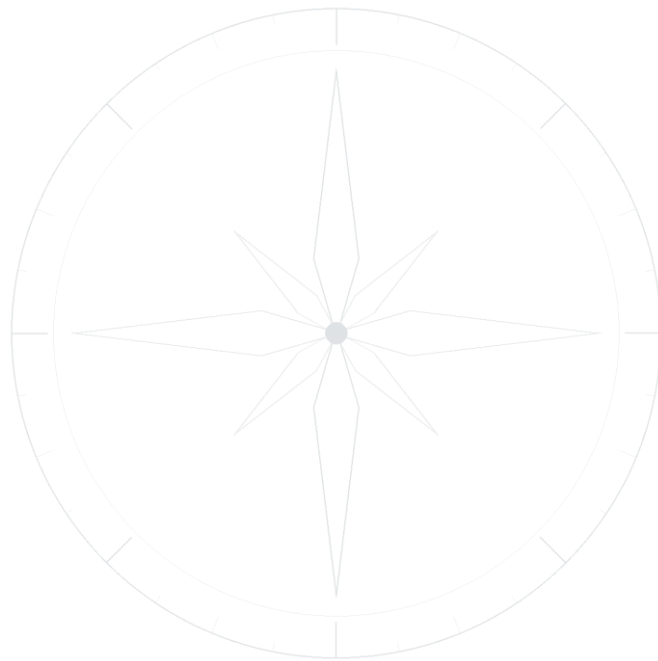
PDCA+ is not a replacement for ISO 31000, NIST 800-128, ISO/IEC 27035, ITIL 4, or any other standard that articulates a particular management system. It is a structural framework for how those standards' disciplines align with each other. Each standard remains the authority for its own subject matter; the disciplines those standards articulate are what bind to PDCA+.

In Annex SL terms: PDCA+ does not displace the High-Level Structure that ISO management-system standards now share. It addresses something the High-Level Structure leaves implicit. The High-Level Structure recognises that management-system standards have common shape — common clauses for context, leadership, planning, support, operation, performance evaluation, and improvement. PDCA+ makes the next move: it recognises that the activities those clauses describe are not properties of the standards that articulate them, but activities that show up across well-joined disciplines. The High-Level Structure organises the standards; PDCA+ describes what happens when the disciplines those standards govern align with each other.

PDCA+ also clarifies a recurring confusion in how PDCA itself is interpreted. Each ISO standard's introduction describes its own PDCA cycle as if the cycle belonged to the standard. The cycle named in ISO 31000's introduction and the cycle named in ISO/IEC 27001's introduction are not the same cycle described twice; they are descriptions of the activities that become visible when each discipline applies the strands to its own subject matter. The activities are common; the subject matter differs. PDCA+ is what the activities look like when more than one discipline's work is observed together.

The practical implication for an organisation is that management-system documentation can be reorganised. Instead of maintaining several manuals, each describing its discipline's own cycle as if that cycle were the structural reality, an organisation maintains one description of the conjunction's substrate and one binding declaration per discipline. The standards remain authoritative for their disciplines' substance; what the documentation describes is how the disciplines participate in the shared lifecycle. The substantive work of each discipline is unchanged; the documentation simply stops claiming that the discipline owns the activities it performs.

For organisations operating across regimes — particularly ISO 27001 alongside NIST RMF, where two complete management-system articulations have to coexist coherently — PDCA+ provides the structural account of how the coexistence works. The disciplines articulated by each regime bind to the conjunction; the conjunction's substrate translates the artefacts that flow between them; the role authority matrix records how named accountabilities carry across both. The two regimes do not need to be reconciled into one. They need to be held coherently alongside each other, and the conjunction is what holds them.



7 References

Origin of PDCA

- Shewhart, W. A. (1939). *Statistical Method from the Viewpoint of Quality Control* (ed. W. E. Deming). Washington, D.C.: Graduate School of the U.S. Department of Agriculture. (Dover reprint 1986.)
- Deming, W. E. (1986). *Out of the Crisis*. MIT Press.
- Deming, W. E. (1993). *The New Economics for Industry, Government, Education*. MIT Press.

PDCA in management-system standards (Annex SL / ISO/IEC Directives Part 1, Annex L)

- ISO 9001 — Quality Management Systems
- ISO/IEC 27001 — Information Security Management Systems
- ISO 14001 — Environmental Management Systems
- ISO 45001 — Occupational Health and Safety Management Systems

Standards typically realised as implementations of PDCA+

- ISO 31000 — Risk Management
- NIST SP 800-128 — Security-focused Configuration Management
- NIST SP 800-53 / NIST RMF — control selection, assessment, authorisation
- ISO/IEC 27035 — Information Security Incident Management
- ITIL 4 — Event Management, Change Enablement, and related practices