An Operating System for the Network Society

HiveMind

Smart by Design

How organizations collaborate intelligently
in a networked, AI-augmented world

Nicolas Bürkler

Book Edition · 2026

HiveMind — Smart by Design

HiveMind

Smart by Design

An operating system for the network society —
how organizations collaborate intelligently
in a networked, AI-augmented world

Nicolas Bürkler

HiveMind — Smart by Design: An Operating System for the Network Society.

First edition · 2026.

The works and sources drawn on in the text are listed in the Bibliography at the end of the volume.

Intelligence is a property of structure,
not of the individual minds.

The thesis of this book

Contents

A Note on This Edition 8

Abstract 9

Overview at a Glance 10

Part One — Foundations

1 Introduction: Why «Smart by Design» 15

2 Theoretical Foundations 22

3 The HiveMind Framework as an Operating System 26

Part Two — The Five Pillars

4 Pillar I: The Policy Doctrine 32

5 Pillar II: Cluster-Based Organization 35

6 Pillar III: Operational Business Architecture 41

7 Pillar IV: Information Governance 44

8 Pillar V: AI Augmentation 48

8a Behavior: The Connective Tissue 51

Part Three — Implementation

9 The Implementation Playbook 57

10 The Maturity Model 60

Part Four — Placing the Framework

11 Network Society, Network Organization 69

12 The Networked Organization: Working Across Company Boundaries 72

13 Conclusion 76

Appendices 78

A Policy-Doctrine Template 79

B Extended Maturity-Assessment Worksheet 82

C Glossary 99

D The Five Pillars as a System 103

E Relationship to Digital Sovereignty as a Service (DSaaS) 105

F Anti-Patterns: What HiveMind Is Not, and Why That Matters 107

Bibliography 110

A Note on This Edition

This is the book edition of the white paper «HiveMind — Smart by Design: An Operating System for the Network Society». It is written for those responsible for transformation, for executive teams, for heads of operations and organization, and for anyone rebuilding the operating model of their organization.

The book is a **universal argument with a Swiss worked case**. The theory is general and travels across jurisdictions. It covers the network society, collective intelligence, run versus build, and complexity-based authority. The concrete instances are Swiss, and are marked as such wherever they appear: firm sizes, regulatory texture, and the sovereignty concern. A reader elsewhere can substitute their own context without the argument breaking.

Throughout, three reference sizes anchor the discussion: a **small** organization of about **15** people, a **mid** organization of about **100**, and a **mid-large** organization of about **4000**. They are spaced roughly an order of magnitude apart on purpose, because the central claim about size is counter-intuitive and only becomes visible across that range.

Abstract

This book proposes HiveMind as a framework for building organizations that are smart by design. They are collectively intelligent through the structure of how they work, not through the unaugmented brilliance of the individuals inside them. The thesis: in a knowledge economy mediated by digital networks and increasingly augmented by artificial intelligence, the competitive frontier has shifted from individual talent to organizational design. The organizations that win are not those with the best people but those whose architecture lets ordinary people produce extraordinary collective results.

Two ideas give the framework its spine. The first is an **operating-system** metaphor: the five pillars are not a list but the components of one machine: a kernel of doctrine, a scheduler of coordination bodies, a file system of information governance, processes-with-permissions in the form of clusters, and AI agents as drivers running on top. The second is a single organizing principle that runs underneath all of it: **authority, and reward, track the complexity a person handles, not the headcount they command.**

HiveMind brings together four strands that have lived in separate literatures: the theory of the network society, the research on collective intelligence and superminds, the practitioner movement around networks of teams, and the young science of hybrid human-AI intelligence. It then adds an operational backbone from information-governance practice. The argument is normative as much as descriptive: organizations that fail to make this transition over the next three to five years will not merely be less competitive; they will be structurally unable to participate in the patterns the network society now sets as its baseline.

Overview at a Glance

The problem. Most organizations were designed for a different era, one of information scarcity, expensive coordination, and physical co-location. None of those assumptions still holds, yet the org chart, the meeting rhythm, the document archive, and the email-based decision path still look like 1995. The result is a productivity ceiling that no amount of hiring and no tool can break, because the ceiling is the design.

The thesis. Intelligence in an organization is a property of its architecture, not of its members. A well-designed organization of average people reliably outperforms a poorly designed one of brilliant people.

The organizing logic, read this before the pillars. Two cuts run through the entire book:

— **Run versus build.** *Build* is the work of authoring the organization, including its doctrine, its cluster design, and its process architecture: high-novelty, fast-changing, done rarely. *Run* is the work of executing the known: high-volume, stable, repeated. The decisive insight is that **build is nearly size-invariant while run scales with throughput.** A firm of 15 and a firm of 4000 with the same product complexity need roughly the *same* build; the larger one simply has far more run, meaning more people serving more clients with the same product. Big organizations stall not because their problems grow but because they over-invest in run and never author the build layer, so every new client adds coordination cost. (This is, at root, a transaction-cost argument: coordination cost, not production cost, is what the design economizes.)

— **Authority tracks complexity, not headcount.** Complexity here means two things together: *intricacy* (how many interacting parts)

and *dynamics* (how fast the ground moves, how much of the work is novel). The width of a person's decision authority is set by the complexity they hold, not by how many people report to them. A small, high-dynamics build line legitimately outranks a large, stable run line. Authority is real but **contestable**: competing complexity claims are arbitrated laterally in the coordination bodies, never self-declared. Compensation follows the same rule as a principle: pay tracks complexity handled, not headcount commanded (the implementation of that is a later-maturity concern, treated lightly here).

The five pillars

Each is annotated build or run, and how each scales:

— **A policy doctrine.** *Build, size-invariant.* The constitutional layer: purpose, principles, rules. Small and dense, not a binder.

— **Cluster-based organization.** *Build to design, run to staff.* Bounded domains, each with an accountable lead, each able to hold both a build line and a run line.

— **Operational business architecture.** *Build at the macro layer, run at the micro layer.* Process documented fractally, lived not filed.

— **Information governance.** *Run-heavy; scales hardest with throughput.* How information is created, classified, retained, disposed. The precondition for everything, above all for AI.

— **AI augmentation.** *The run-absorber.* Not a gadget bolted on, but governed colleagues that take over low-dynamics run work and raise the complexity, and therefore the authority, of the humans who direct them.

The path in three phases

Year 1 is doctrine and foundation (almost entirely build). Year 2 is architecture and adoption. Year 3 and beyond is augmentation and maturity. A five-level maturity model (*uncontrolled* → *emerging* →

forming → *operational* → *proactive*) locates an organization and sets its next step.

Part I

Foundations

This part lays the foundation. Chapter 1 explains why «smart by design» is the right idea and introduces the two cuts, run/build and complexity-based authority, that organize the book. Chapter 2 credits the literatures HiveMind synthesizes and names its intellectual ancestors honestly. Chapter 3 presents the framework as a single machine through the operating-system metaphor.

Chapter 1

Introduction: Why «Smart by Design»

1.1 The beehive as metaphor

The word «hive mind» usually arrives carrying its science-fiction meaning: a swarm of mindless drones with no inner life, fused into a single will and stripped of any self. (Star Trek fans will think of the Borg, and of Seven of Nine slowly recovering her individuality after being cut off from the collective.) That is emphatically *not* the image this book intends. The real beehive is almost the opposite, and it is the better metaphor precisely because it is real.

Start with the individual bee, because the science is surprising. A honeybee is not a mindless automaton. It has a sophisticated, time-aware memory: trained bees reverse a learned preference depending on the time of day and on whether they are heading out to forage or returning home, in effect using context to decide what to do and when (Zhang et al., 2006). Each bee is highly capable at its own job. The colony is built on competence, not on the absence of it.

What turns those capable individuals into something greater is not a collective mind that overrides them but a communication system of remarkable precision. A returning forager performs a waggle dance whose angle and duration encode the direction and distance of a food source, and the colony reads these signals to allocate effort across a whole landscape, optimizing where to send foragers without anyone directing the whole (von Frisch, 1967). No single bee holds the colony's strategy. The intelligence lives in how capable individuals share what they know through a structured, shared channel, so that a hive of forty thousand can navigate a landscape, allocate labor between foraging and

brood care by season, defend itself, choose the site of a new nest by a procedure a political scientist would recognize as quorum democracy, and hold a stable internal temperature across a winter no individual bee could survive.

That is the whole metaphor, and it maps directly onto the argument of this book. Collective intelligence comes from capable parts plus a structure that composes them: capable people, joined by a well-designed organization built to optimize how information and data move between them. The bee is not the dumb cog of the science-fiction trope; it is the skilled worker whose contribution is multiplied by an extraordinary system of coordination. That is exactly the bet HiveMind makes about people inside a well-designed organization.

The beehive is a metaphor, not a model; the biology is not the point. The point is that collective intelligence is real, measurable, and arises from the structure in which capable actors interact and communicate rather than from any single actor's brilliance. This is the central empirical finding the book builds on. Because the book leans on it hard, it is worth stating its limits plainly up front (see 1.2).

What is new in 2026, and what makes the metaphor productive rather than decorative, is that the preconditions for structured, transparent, machine-augmented collective intelligence among humans now actually exist. Until recently humans coordinated through hierarchy (someone tells you what to do) or markets (prices tell you what to do). A third mode, stigmergic coordination through a shared, structured environment that everyone reads and writes, where the traces left behind guide the next action, was a theoretical possibility but a practical impossibility beyond a few dozen people, because maintaining the shared environment cost too much. That cost has collapsed. A modern collaboration platform gives even a large organization a shared, structured, versioned environment; add an AI layer that can read, summarize, and update it, and coordination cost

falls another order of magnitude. The infrastructure for stigmergic human organization is, for the first time, off the shelf. The question is no longer whether such organization is possible, but why so few have built it.

1.2 The thesis, and an honest word on the evidence

> A well-designed organization of average people will reliably outperform a poorly designed organization of brilliant people.

The support for this comes from the research on collective intelligence: groups have a measurable «c factor» that predicts performance across many tasks, correlates only weakly with members' average IQ, and correlates strongly with structural features: turn-taking, social sensitivity, the design of the information environment, and increasingly the deliberate inclusion of machine participants.

The honest caveat. That evidence is largely drawn from laboratory studies of small groups on short tasks. The leap from «small groups in a lab» to «a 4000-person firm over three years» is an inference, not a proven result. The book makes that inference deliberately, and here is why it is defensible: the *mechanism* the lab work identifies, that coordination structure, not member brilliance, drives collective output, is exactly the mechanism that operates at organizational scale, where structure is even more decisive because informal coordination breaks down sooner. The lab evidence is suggestive, not dispositive; the organizational claim should be read as a well-grounded engineering bet, not a theorem. The rest of the book is the design that follows if the bet is right.

If collective intelligence is structural, it can be constructed. A leader's task is not to be the smartest person in the room, a heroic idea organizational research keeps puncturing, but to design the room: the

doctrine, the cluster boundaries, the architecture, the information environment, the augmentation pattern, the culture. Get it right and the organization is smart almost regardless of who is in it; get it wrong and no amount of talent saves it.

1.3 The organizing logic: run vs. build, and authority by complexity

Two cuts organize everything that follows. They are introduced here and then threaded through every pillar.

Run versus build. *Build* is the work of authoring the organization: writing the doctrine, designing clusters, drawing the macro architecture, deciding how AI participates. It is high-novelty, fast-changing, and done rarely. *Run* is the work of executing the known: classifying the thousandth document, serving the hundredth client through an established process. It is high-volume, stable, and repeated. The two demand opposite conditions: build needs slack and uninterrupted attention, while run needs flow and consistency. The same person rarely does both well at the same time, because an operative queue destroys the slack that initiating work requires.

The consequence that reorganizes the book: **build is nearly size-invariant; run scales with throughput.** A 15-person firm and a 4000-person firm with the same product complexity need roughly the same doctrine and the same architecture. The build is the same. The 4000-person firm simply has vastly more run: more people executing the same known work for more clients. Complexity does not multiply with headcount; *throughput of the known* does. This is why large organizations stall. They are not solving harder problems than small ones; they are serving more clients at the same product complexity, and they cope by adding run, more managers and more coordination overhead, while never authoring the build layer that would give all that run its rails. Friction compounds: run grows with clients, build stays flat and

ungoverned, and the organization slows by a constant fraction each quarter until a network-fit competitor has pulled away.

Authority tracks complexity, not headcount. If build and run are the two kinds of work, the next question is who gets to decide what. The classic answer, that authority accrues to whoever manages the most people, is exactly what produces empire-building and bloated middle layers. HiveMind inverts it: the width of a person's decision authority is set by the **complexity** they handle, where complexity is *intricacy* (interacting parts) and *dynamics* (speed of change, proportion of novel work) together. A small build line working at high dynamics legitimately holds wider authority than a large run line executing stable, known work at volume. Decisions are still made at the lowest competent level, but «competent» means competent for *this complexity*, and the boundaries of a run owner's authority are narrower not because they are junior but because their domain is, by design, more bounded. Both are real domain authority; they differ in width, not in legitimacy.

Two guards keep this from becoming its own politics. First, complexity is harder to measure than headcount, which is precisely why organizations default to headcount, so complexity-based authority must be **contestable**: competing claims are arbitrated laterally in the coordination bodies against a shared rubric, never self-declared, or the prize simply goes to whoever argues most fluently. Second, the same logic extends to reward, since **compensation tracks complexity handled, not headcount commanded**, so that pay does not quietly reintroduce the hierarchy the structure removed. The principle is stated here and in the doctrine; its detailed implementation is a later-maturity concern and is treated lightly (Chapter 5, Chapter 10).

1.4 What «smart by design» means in practice

It means intelligence is built into the organization rather than squeezed out of individuals on demand:

— **Knowledge lives in the shared environment, not in heads.** What one person knows, the organization knows.

— **Decisions are made at the lowest competent level** (competent for the complexity at hand).

— **Processes are documented, lived, and revised;** improvisation is reserved for genuinely novel problems.

— **The environment carries the coordination load,** like the pheromone trail of an ant colony.

— **AI agents are designed in, not bolted on,** with an explicit split between human, AI, and hybrid work.

The result is robust (the loss of an individual is recoverable), scalable (onboarding is the structure), transparent, adaptive, and augmentable.

1.5 The argument from the moment

Beyond the structural case there is a timing case. Three signals make it concrete.

The hybrid-intelligence signal. A growing body of research shows the next productive frontier is neither AI replacing humans nor humans using AI as a tool, but hybrid systems designed as integrated units of collective intelligence. The widely noted 2024 meta-analysis is blunt: human-AI combinations beat humans alone or AI alone *only when the integration is designed*. Untuned deployments, on average, add nothing, and sometimes less.

The augmentation signal. Large-scale research on AI return in 2025 found that organizations which added AI to existing processes got efficiency; organizations which *redesigned around* AI got transformation. Value is unlocked by organizational design, not by tools.

The sovereignty signal (universal, with a Swiss instantiation). The regulatory and sovereignty environment is tightening across jurisdictions: information practices that were tolerable in 2020 are liabilities by 2026, and organizations without a doctrine, without classification, without governance of their AI use are accumulating compliance debt that will come due. *In the Swiss context specifically*, this shows up as the exclusion of CLOUD-Act-bound providers from new health-data infrastructure, the eCH standardization work, evolving cantonal positions on digital sovereignty, and the reach of the EU AI Act. A reader in another jurisdiction can substitute their own instruments; the mechanism is the same.

These are one transition, not three: from organizations that *use* digital tools to organizations that *are* digital systems of collective intelligence.

Chapter 2

Theoretical Foundations

HiveMind invents none of its foundations. This chapter makes the synthesis visible and, more honestly than the white paper did, names its ancestors, so the reader can trace each pillar to the body of work it descends from. Full references are in the Bibliography.

2.1 The network society

The canonical formulation holds that the dominant functions of the information age organize around networks, which form the new social morphology of society. Three claims are load-bearing: the shift from a *space of places* to a *space of flows* (work happens in the flow, not the building); the *network enterprise* as the right unit of analysis rather than the firm; and inclusion/exclusion in networks as the structure of power, which makes information governance strategic, not clerical. More recent work adds the *algorithmic turn* the original theory did not anticipate: networks are no longer only conduits but increasingly participants that generate content and make decisions.

2.2 Collective intelligence and superminds

This supplies the operational substance: group intelligence is real, measurable, and designable; superminds come in four forms (hierarchies, democracies, markets, communities), and HiveMind is a deliberate hybrid (hierarchy within clusters, community across them, small markets where useful); computers extend superminds rather than replacing them; and the «design genome» of a collectively intelligent system can be mapped along what it does, who does it, why, and how. (See 1.2 for the honest limits of this evidence.)

2.3 The network of teams and its nearest modern cousin

The practitioner literature documents the displacement of functional hierarchies by team-based structures: empowered teams, network of teams, squads and tribes. HiveMind's cluster model is one instance of this pattern, distinguished by named accountable leads, defined coordination bodies, and a doctrine that bounds autonomy. **The closest contemporary relative is Team Topologies** (stream-aligned, platform, enabling, and complicated-subsystem teams, plus defined interaction modes). HiveMind's clusters map closely onto stream-aligned teams, its IG/architecture functions onto platform teams, and its build lines onto enabling work, and the comparison is far more apt than the usual one to Holacracy. (The contrast with Holacracy, Teal, and agile is taken up in Appendix F.)

2.4 Policy doctrine and operational architecture: the deeper lineage

This is where the white paper was thinnest about its debts, and where the strongest ancestors live:

— **Polycentric governance.** Elinor Ostrom's design principles for governing a commons (clearly bounded domains, rules made by those who live under them, graduated and locally arbitrated enforcement, and nested layers) are almost exactly HiveMind's doctrine, domain authority, and lateral coordination. This is the single most important foundation to read: it gives the «not flat, not centralized» position a rigorous, empirically grounded basis.

— **The viable-system view.** HiveMind's fractal macro/meso/micro architecture, and its insistence that each level be viable in its own right

while nested in the next, is cybernetic organizational design in Beer's lineage.

— **Matching structure to information flow.** Galbraith's information-processing view holds that structure should match the information flows the organization must process, and Conway's law observes that systems come to mirror the communication structures that build them. Both ground the claim that architecture should follow flows, not legacy hierarchy.

— **The economics of coordination cost.** The macro claim that «coordination cost has collapsed» is a transaction-cost argument, the line that runs from Coase to Williamson: the boundary and shape of the firm track the cost of coordinating, and when that cost falls by orders of magnitude the efficient structure changes.

A policy doctrine, in this lineage, is a small, dense constitutional document (Prime Directive, a handful of principles, rules of the game, a revision process), and operational business architecture is its practical complement, built bottom-up by those who do the work.

2.5 Hybrid human-AI intelligence and its forebears

The youngest foundation studies what happens when AI joins groups as a participant, not a tool. The 2024 meta-analysis of 106 studies finds that naive integration yields no gain while *designed* combinations yield substantial gains, especially on creative work; a companion 2024 model frames hybrid intelligence as a multilayer network (physical, informational, cognitive) that produces composite intelligence only when the layers are aligned. The practical implications, that capability is not productivity, that the unit of analysis is the hybrid group, and that curation precedes augmentation, make AI the *last* pillar in sequence. **The vision itself is old and worth crediting:** Licklider's «man-computer symbiosis» (1960) and Engelbart's «augmenting human intellect» (1962) named this destination

sixty years before the tools arrived. HiveMind's «AI as colleague» is their idea, now finally buildable.

2.6 The mechanisms HiveMind relies on by name

Three mechanisms the book uses implicitly deserve their names: **stigmergy** (coordination through traces left in a shared environment), **transactive memory** (the formal account of «knowledge lives in the environment, not in heads», where who-knows-what becomes a property of the group), and the **tacit-to-explicit knowledge spiral** (the bottom-up «document your own work» method is exactly this conversion of tacit practice into explicit, shareable structure).

2.7 What HiveMind adds

Each foundation alone is incomplete: network theory is descriptive not prescriptive; collective-intelligence research gives limited operational guidance; the network-of-teams literature is silent on information governance and AI; the doctrine tradition is usually divorced from architecture; the hybrid literature is only beginning to address enabling design. HiveMind's contribution is the integration: one framework that treats all of this as a single designed system, organized by the operating-system metaphor and the run/build and complexity logics introduced in Chapter 1.

Chapter 3

The HiveMind Framework as an Operating System

3.1 The operating-system metaphor

The subtitle is not decoration. An organization, like a computer, runs many concurrent activities on shared resources, and needs a layer that decides what runs, with what permissions, against what rules. The five pillars are the components of that operating system, and seeing them as one machine is what makes «five» feel necessary rather than arbitrary:

OS component	HiveMind pillar	What it does
Kernel	Policy doctrine	The small set of rules everything else runs under; mediates every conflict the higher layers cannot.
Processes, with permissions	Cluster organization	Bounded units of execution, each with an owner and a defined scope of what it may touch.
Scheduler	Coordination bodies	Allocates attention and resolves contention between clusters laterally, on a cadence.
File system	Information governance	How information is named, structured, classified, stored, retained, and disposed.
Drivers / daemons	AI augmentation	Governed agents that run continuously under the kernel's rules, doing defined work and reporting to named humans.
Hardware	DSaaS (companion)	The sovereign, governed substrate the whole system runs on (Appendix E).

The metaphor also disciplines the design. A kernel must be small (a 60-page doctrine is a bloated kernel). Processes must not reach into each other's memory without going through the scheduler (clusters coordinate through bodies, not raids). You do not install drivers before the file system is mounted (AI comes after governance). The OS frame is used lightly through the book, but it is the binding that turns a list of five into a system.

3.2 The five pillars, annotated by run/build and scale

Pillar	Build or run	How it scales 15 → 100 → 4000
1. Policy doctrine	Build; size-invariant	Same doctrine at every size; only the ratification and review apparatus thickens.
2. Cluster organization	Build to design, run to staff	15: still 5–8 clusters, but with heavy overlap, so one person may lead several and also execute. 100: a similar cluster count, but the coordination layer becomes the binding constraint and leads delegate. 4000: clusters become divisions and the model recurses fractally.
3. Operational architecture	Build (macro) + run (micro)	Macro map is similar at all sizes; micro documentation and automation grow with throughput.
4. Information governance	Run-heavy	15: one catalog, one part-time owner. 100: an IG owner and a real catalog. 4000: an IG cluster, audits, tooling.
5. AI augmentation	The run-absorber	The larger the run surface, the more AI displaces, and the more the human role shifts to directing agents.

The table is the practical face of the run/build claim: read down the «build» rows and the work barely changes across a 250× range of headcount; read down the «run» rows and it grows steadily. That asymmetry is the whole argument about size.

3.3 Properties the framework produces

Robustness, transparency, scalability, adaptivity, augmentability, and accountability, each a consequence of the structure rather than of

heroic individuals. (The full treatment, and what the framework explicitly is *not*, is consolidated into the anti-patterns material in Appendix F, to avoid saying it twice.)

Part II

The Five Pillars

The five pillars are the substantive core. Each chapter treats one pillar: what it is, where it sits on the run/build line, how it scales across 15 / 100 / 4000 people, how it connects to the others, and how it fails. The order is the order of dependency: doctrine first, AI last. Behavior, which the white paper buried inside the cluster chapter, is now its own chapter (8a) at the end of this part, because it is the connective tissue all five pillars depend on.

Chapter 4

Pillar I: The Policy Doctrine

> **Build / run · scale**
>
> Doctrine is pure *build* and almost perfectly *size-invariant*: a 15-person and a 4000-person firm with the same product complexity need the same small constitution. Only the ratification and review apparatus thickens with size.

4.1 What a doctrine is, and what it is not

A policy doctrine is the constitutional layer, the kernel in the operating-system metaphor of Chapter 3. It is the small, dense document that defines what the organization exists to do, by what principles it operates, how decisions are made, and how it changes. It is not a policy manual: a manual is thick, specific, and tries to anticipate every case; the doctrine is thin, abstract, and trusts the reader to apply principles to cases. The thick manual fails by going unread and being quietly circumvented; the thin doctrine fails only if it is wrong in principle, and then the failure is visible and fixable. Nor is it the vision/mission/values of a website: those are aspirational, the doctrine is operational. It must be enforceable: when a real decision arises, the doctrine must produce an answer or name who is authorized to.

> **A useful test**
>
> Take a hard decision the organization faced in the last twelve months. Would the doctrine, honestly applied, have helped resolve it? If yes, it is doing its job. If no, it is underdeveloped or decoration, and it must then be revised, updated, or amended until it does.

4.2 The structure of a HiveMind doctrine

A doctrine contains at least: a **Prime Directive** (one sentence on what the organization exists to do); a small set of **principles** (five to twelve axioms, each action-guiding, falsifiable, justifiable, durable); **decision rights**; the **information-governance** and **AI** positions in summary; the **change process**; and the **behavioral expectations** (treated fully in Chapter 8a).

Two principles are new to this edition and belong in every HiveMind doctrine, because they are what stops the old hierarchy from creeping back through the side doors of title and pay:

— **Authority tracks complexity, not headcount.** A person's decision authority is as wide as the complexity (intricacy plus dynamics) they hold, not as wide as the number of people they manage. Authority is real within its domain and **contestable** through the coordination bodies; it is never self-declared.

— **Compensation tracks complexity, not headcount.** Reward follows the same rule as authority. Acquiring reports is not a raise. People-leadership is itself a complexity domain, paid for the difficulty of developing and coordinating humans rather than for the count, and it sits alongside technical, strategic, and client complexity rather than above them.

Example principles organizations typically also adopt: **Once-Only**, **Open by Default**, **Structure before storage**, **Domain authority is real**, **Lateral coordination, not escalation**, **Clarity over consensus**, **Bad news ages badly**, **AI outputs require human accountability.** These are examples; each organization derives its own small, memorable set.

4.3 What a doctrine does for the organization

It **liberates** (it removes the burden of re-litigating first principles, since clear rules free people to act), it **aligns** (teams applying shared principles to local cases stay coherent without central coordination, the stigmergic property at the doctrinal level), and it **scales** (a new cluster or product line does not require a new operating model; the doctrine applies to the new context).

4.4 Doctrine and architecture: the synergy

A doctrine without architecture is air; an architecture without doctrine is mechanism without purpose. The doctrine says «decide at the lowest competent level», but if no one knows where the levels are or who is competent for what, the principle produces no action. The two are interlocked: strategic coherence emerges only where both are in concert.

4.5 The doctrine as a living document

Reviewed at least annually, versioned, and visibly used, meaning referenced in decisions, meetings, and escalations. If a year passes without a single reference, the doctrine is dead even if the file exists. Decisions that violate it are reversed, or the doctrine is changed; letting a violation stand while the doctrine pretends otherwise is the one illegitimate option.

4.6 What goes wrong with doctrines

Too long (a 60-page doctrine is a bloated kernel, so aim under 20); too abstract (principles no one could violate); too consensual (every edge sanded off); not enforced (decays within a year); frozen (never changed, ages out); or written by consultants (a constitution its authors did not write does not bind).

Chapter 5

Pillar II: Cluster-Based Organization

Build / run · scale

Designing the cluster map is *build*; staffing and running the clusters is *run*. **15:** typically still five to eight clusters, because the number of clusters tracks the kinds of work to be owned rather than headcount; people simply overlap, with one person leading several and also executing. **100:** a similar cluster count, but the coordination layer becomes the binding constraint, and leads delegate. **4000:** clusters become divisions and the whole model recurses, with clusters inside clusters, each with its own doctrine-consistent local structure.

5.1 The structural problem

The classic functional pyramid has been declared obsolete for thirty years, yet most organizations still run one, or a thin agile veneer over it, because the alternatives have failed at scale: pure self-management collapses into ambiguity, Holacracy's process overhead consumes the autonomy it promised, matrices confuse reporting, and flat structures hide informal hierarchies that work worse than the formal ones they replaced. HiveMind's answer is the cluster.

5.2 The cluster model

A cluster is a bounded operational domain with a defined scope (what is inside and, crucially, outside), an accountable lead, a team, and explicit interfaces to other clusters. It is organized by *outcome*, not by *function*, that is, by the bounded domain of decisions and execution that produces a particular result.

Build lines and run lines inside the cluster. Build is a *core function*, not a separate caste living off to the side. It is present throughout the organization as a **build line inside clusters**: every cluster can hold both a build line (authoring and improving how the cluster works, at high dynamics and narrow headcount) and a run line (executing the established work at volume). A small cluster may be all run with a sliver of build; a fast-moving one may be mostly build. This is how «build is size-invariant» is realized concretely: the build lines across the organization sum to roughly the same effort whether the firm is 15 or 4000, while the run lines grow with throughput.

5.3 The operating principles

— **One accountable lead per cluster.** Pluralized accountability is a euphemism for none.

— **Domain authority is real, not ceremonial**, and its *width is set by complexity* (Chapter 1), so a build line's authority over an ambiguous, fast-moving domain is wider than a large run line's authority over stable, known work. Both are genuine; they differ in width, not legitimacy.

— **Cross-alignment through lateral coordination, not escalation.** Two clusters that must align use a governing body, not a manager. The bodies are also where competing complexity claims, for authority and for pay, are arbitrated, so complexity-based authority cannot be self-declared.

— **Decisions at the lowest competent level** (competent for *this* complexity).

— **Clarity over consensus, progress over perfection.**

5.4 The cluster lead as mode-switcher

The cluster lead is not a pure creator and not a pure operator. The role specifically demands the ability to *switch modes*: to author the build line (invent and revise how the cluster works) and to care that the run line executes reliably. Pure creators who cannot sustain make poor leads; pure sustainers who cannot initiate make poor leads. The lead is the person who can both build the rails and ensure the trains run, which is why lead selection looks for range rather than a personality type.

How this changes with size. Mode-switching is the small-organization case. In a firm of about 15, the cluster lead genuinely does both, authoring the build line and keeping the run line honest within the same week. Past about 100 people that stops being realistic, because the load is too high for one person to hold both modes well. The more realistic pattern at scale is a pair: the cluster lead owns the build line, which is the high-complexity work of authoring and improving how the cluster works, and a close partner, a deputy or an operations lead, owns the run line. The lead still sets direction across both, but the day-to-day running is held by someone whose strength and preference is the sustaining mode. The principle does not change; only the division of labor scales with size.

5.5 Cluster definition: scope, interfaces, exclusions

The hard work is specifying the scope: inclusions, **exclusions** (what one might assume belongs but explicitly does not), and interface

points. The exclusion section is where most definitions are deficient and where silent conflict is born; writing it forces the conversation that surfaces disagreement before it becomes an operational problem.

5.6 The coordination layer

Above the clusters sit governing bodies, covering leadership, product, operations, and development and innovation, each with named permanent participants, a named lead, a defined cadence, decision authority, and documented outcomes. They are the *scheduler* of the operating system: they allocate attention and resolve contention laterally, replacing escalation upward, and they are the venue where complexity claims are adjudicated.

Body	Mandate	Cadence
Leadership body	All cluster leads. Cross-cutting decisions, priorities, resource conflicts, doctrine changes, complexity/ authority arbitration.	Biweekly
Product body	Product/ engineering + sales + strategy. Roadmap, demand prioritization, build-or-buy.	Monthly
Operations body	Backbone + strategy + rotating representation. Budget cycles, hiring plans, compliance.	Monthly
Dev body (development & innovation)	Information management + engineering + backbone. Information architecture, AI deployment, governance evolution, and new capabilities.	Monthly

5.7 Why the cluster model is not flat

There is clear hierarchy *within* a cluster (the lead has authority, the team executes); what is lateral is the relationship *between* clusters. No cluster sits structurally above another; the leadership body is a body of peers, its lead a *primus inter pares*, not a CEO. This non-flat-but-lateral shape preserves accountability (each cluster owned by a named person) while distributing authority (no cluster owned by another), and the complexity rule, not headcount, is what ranks influence across the peer body.

5.8 Failure modes

Clusters become silos; bodies become rubber stamps or, oppositely, escalation venues; leads revert to «managers» and bottleneck execution; the doctrine is forgotten and every decision becomes a fresh negotiation; or, the failure this edition most wants to prevent, **headcount quietly recurs as the basis of authority and pay**, and empire-building returns. The symptom of the last is leads competing to absorb people and budget rather than to hold harder problems.

Chapter 6

Pillar III: Operational Business Architecture

Build / run · scale

The macro map (value streams and how clusters own them) is *build* and nearly size-invariant; the meso/micro documentation and the automation that rides on it are *run* and grow with throughput. **15:** a one-page macro map and a handful of lived workflows. **100:** documented meso processes per cluster. **4000:** a documentation discipline, with the micro layer increasingly executed and maintained by AI.

6.1 The case for documented work

Most organizations run on implicit knowledge, which is tolerable at five people, untenable past twenty, and disastrous past a hundred. Implicit operation does not survive new members, lost veterans, new work, or the arrival of AI; each exposes the implicit knowledge as a deficit. (This is, in the lineage of Chapter 2, the conversion of *tacit* knowledge to *explicit*, the knowledge-creation spiral made operational.)

6.2 What an operational architecture is

A layered, fractal, navigable representation: value streams, the processes within them, the roles that perform each step, the information that flows through, the systems that support it. Fractal because navigable at several zoom levels; lived because created by those who do the work, not delivered by a consultancy and filed. This is the *Viable System Model* lineage: each level viable in itself, nested in the

next. The test is not whether it exists but whether someone consulted it this morning.

6.3 The bottom-up method

Resist the top-down temptation (interviews, beautiful diagrams, ignored). Begin with those who do the work: **(1)** everyone documents what they actually do; **(2)** teams aggregate into workflows, surfacing overlaps and gaps; **(3)** the organization connects workflows end-to-end; **(4)** the documented processes are *activated* as the operating mode, so the documentation becomes the script, not its description.

6.4 The fractal property

Macro (the leadership view: value streams and clusters), meso (the cluster-lead view: processes with owners and gates), micro (the executor view: tasks, tools, artifacts), engine room (screens, keystrokes, API calls, where AI agents operate). Conway's Law is the warning here: the architecture will come to mirror the communication structure that builds it, so the build of the architecture is itself a structural act.

6.5 Documented workflows replace synchronous coordination

Most meetings disappear (their legitimate function, namely new options, genuinely novel problems, and social cohesion, remains; the administration of established workflows falls away). Most coordination becomes *stigmergic*: the team reads a project's state in the shared environment rather than convening to ask. The result is more substantive contact with colleagues' work and less performative co-ordination.

6.6 Discussions where the work lives

Comments, threads, and decision records live on the artifact, not in a parallel channel or email. Decisions in email are invisible to anyone off the thread; decisions in the workflow are visible to all who need them, including future hires and the AI agents that need the decision context to act.

6.7 Direct editing with clear ownership

A document has a state (draft, review, final, archived) and a permission per state: fast iteration early, careful gating late, and clear authority at each transition.

6.8 The architecture as foundation

Without it, information governance is unanchored, AI has no defined targets, onboarding is oral, continuity is fragile. With it, each becomes tractable. The architecture is the condition for the other pillars, not optional and not deferrable.

Chapter 7

Pillar IV: Information Governance

Build / run · scale

Designing the scheme and the catalog structure is *build*; classifying, retaining, and governing information at volume is *run* — and this is the pillar that scales *hardest* with throughput. **15:** one catalog, one part-time owner. **100:** a named IG owner and a real, maintained catalog. **4000:** an IG cluster, quarterly audits, tooling, and AI-assisted classification.

7.1 The strategic case

Information governance is usually framed as hygiene, the domain of records managers and retention schedules. HiveMind treats it as strategic: in a network society the information environment is the *substrate* of collective intelligence, and stigmergic coordination works only if that substrate is governed: structured, classified, retained, accessible, and trustworthy. An ungoverned environment is not neutral but deteriorating. AI amplifies the stakes by an order of magnitude: AI on a governed environment is powerful augmentation; AI on an ungoverned one is a multiplier of disorder, with apparent authority and at scale. Governance is the *file system* of the operating system, and the precondition for AI, for institutional memory, and for regulatory defensibility from 2026 on.

7.2 The principles

Once-Only; Open by Default; Structure before storage; Single source of truth; Connection rather than silos; Traceability and versioning; Consistent nomenclature.

7.3 The classification scheme

Small and workable — most begin with four levels: **Public**, **Internal**, **Confidential**, and **Secret**, each with defined handling (residence, access, transfer, retention, disposal) and each an input to access control and to AI policy. More than five or six levels yields a scheme no one follows.

7.4 The lifecycle

Creation/capture, active use, reference, archival, and disposal, each with defined responsibilities and retention, and *enforced*: at end-of-period the artifact is actually deleted. Records of records-management significance get that discipline as a specialization within governance.

> **Retention as cleaning by design**
>
> Records management and retention are the *cleaning* function of the information environment. The more the discipline is embedded in daily workflow and built in by design, rather than run as an occasional purge, the less clutter the environment accumulates in the first place. A place that is tidied continuously, by design, never needs the dreaded big clean-up, and it stays trustworthy enough for both people and AI to rely on.

7.5 The information catalog

A structured, living inventory of essential resources: metadata, classification, lifecycle status, and ownership. It is the map of the environment and an input to every governance decision.

> **The catalog as a fact-based steering instrument**
>
> The maturity question is not «how good are we?» but «what does our catalog say objectively about our governance maturity?» The better the catalog is maintained, the higher the measured maturity rises — which turns an IT inventory into a steering instrument.

7.6 AI in the information environment

The most consequential AI decisions are governance decisions: which tools are permitted, which classifications may be processed by which tools, who is accountable for outputs, how use is logged. The artifact is a simple matrix of classification levels × tools (permitted / conditional / prohibited), consulted in the moment and reviewed quarterly. A positive, operational AI-use policy replaces the uncontrolled reality of staff pasting documents into public services.

7.7 Roles

Information owner (per resource class), information-governance lead (one per organization), data-protection officer (where required), records manager (where obligations exist), system owner (per major system). In a 15-person firm these collapse onto one or two people; what matters is the named accountability, which is exactly what scales into an IG cluster at 4000.

7.8 Governance and the cluster model

Governance is owned by a cluster, typically «information management», which holds the doctrine for information, maintains the catalog, convenes the Dev body (development and innovation), and audits; it does not execute on behalf of others. Each cluster governs its

own information under the doctrine. This is «domain authority is real» applied to the governance function.

Chapter 8

Pillar V: AI Augmentation

Build / run · scale

AI is the *run-absorber*: it takes over low-dynamics, high-volume run work. The larger the run surface, the more AI displaces, and, critically, the more the surviving human role shifts from *doing* the run work to *owning and directing* the agents that do it, which *raises* the complexity, and therefore the authority and pay, of run roles over time. **15:** a few curation agents. **100:** agents across several workflows. **4000:** augmentation is the default and directing agents is a general competence.

8.1 From tool to colleague

The 2023/24 framing of «AI as productivity tool» has expired. By 2026 the frontier is «AI as colleague», meaning AI as a participant in the workflow, not a tool a worker picks up. From this: the unit of analysis is the hybrid group; AI participants have roles, not uses; the human-AI mix is a designed variable; workers learn to *direct* AI; and governance is structural, since AI is bound by the same doctrine and the same behavioral expectations as human workers. In the operating-system metaphor, agents are *drivers*: they run continuously under the kernel's rules and report to named humans.

8.2 Where agents fit

For each process: which tasks are pure-human (judgment, accountability, social presence), pure-AI under oversight (routine, well-defined), hybrid (different parts to each), or AI-assisted human (human-led, real-time AI support). The categorization is explicit and migrates as capability grows.

8.3 Workflow composition: the human-AI mix

The decisive planning variable is not the model but, per workflow, the *composition*: capacity (how much AI could competently do), oversight (how much human steering it demands), the optimal allocation today, and how it shifts as capability advances. Composition is at once a budget variable (plan AI capacity like headcount) and a governance variable (it sets how much human review the outputs get and where accountability sits).

8.4 Directing AI: the new core competence

Building, delegating to, and supervising agents is a role that did not exist three years ago and will describe most knowledge workers within five. Its competences, namely task decomposition, specification, review, iteration, and accountability, are trainable and not yet widespread; a HiveMind organization develops them deliberately. **This is the humane core of the run/build story:** as agents absorb repetitive run work, the human who directs them is doing higher-complexity work than before, so the run role rises rather than hollows out. Run, owned and augmented, is not drone work.

8.5 Curation before analysis

Two stages, in order. **Curation** agents structure the environment (classify, generate metadata, detect duplicates, summarize, link, flag). **Analysis** agents draw insight from the structured environment (answer, recognize patterns, support decisions). Analysis on curation produces reliable insight; analysis without curation produces fast, plausible-sounding garbage. Deploy serious analysis only once the environment is curated to the operational level.

8.6 AI governance integrated with information governance

«May I summarize this with a public AI service?» is a classification, transfer, residency, confidentiality, and retention question at once. AI policy and classification scheme are designed together; the matrix of 7.6 is the operational artifact.

8.7 Through the supermind lens

The organization as a whole is a hybrid composition of human and AI participants. Productive capacity is no longer headcount but human × AI; the org chart no longer fully describes the organization (clusters hold human members *and* named AI participants, the latter reporting to human roles); and the organization is cognitively more capable than its members combined, because collective intelligence is emergent, not additive.

8.8 What goes wrong

Augmentation without governance; without curation; without role redesign (AI bolted onto an existing role, which becomes hybrid by accident); without accountability (no named human behind an output, the highest regulatory exposure); without competences (workers use but do not direct AI); or as theater (a board-deck deployment, not a real change in the work).

Chapter 8a

Behavior: The Connective Tissue

The white paper buried behavior inside the cluster chapter. It earns its own chapter, because the structure, comprising doctrine, clusters, architecture, governance, and AI, only produces collective intelligence if people behave in particular ways. Behavior is not a sixth pillar; it is the connective tissue without which the five do not transmit force. (Whether it is «emergent» or «designed» is a false choice: it is designed in the sense that it is named, expected, and enforced, and emergent in the sense that it must be lived, not decreed.)

8a.1 The operating behaviors

These are not values or personality traits but operating requirements, the price of participation:

— **Ownership over compliance.** Care about results, not box-ticking; go beyond the strictly required where it serves the organization.

— **Proactive communication.** Bad news ages badly; whoever knows something is going wrong raises it at once, not at the next scheduled meeting.

— **Direct engagement, constructive intent.** Leads challenge one another when they disagree; challenge is welcome, passive non-engagement is not.

— **No turf behavior.** Domain authority serves speed and clarity, not insulation; clusters are not fiefdoms.

— **Commitment after the decision.** Vigorous disagreement before; once decided, all carry it, without silent reservation or public undermining.

8a.2 Two modes of work, and matching people to them

The run/build cut is also a cut in the kind of attention work demands. *Build* (initiating) work needs slack, tolerance for ambiguity, and uninterrupted divergent attention; *run* (sustaining) work needs flow, consistency, and convergent attention. The two are hard to do at once: give an initiator a full operative queue and their initiating capacity collapses, not because of disposition but because the queue destroys the slack initiating requires.

So the organization should **match people to modes by demonstrated strength and stated preference**, and let the match be reversible. Some people thrive on steady, high-volume, well-defined execution and find open-ended ambiguity draining; others are the reverse. Letting people self-sort toward the mode they are good at is both humane and productive. This is a claim about *work design and fit*, grounded in observable behavior and preference, and deliberately **not** a claim that a personality instrument sorts people into fixed types. People contain both modes, shift across a career, and a hard typology that gated roles would be brittle, of dubious validity, and would quietly recreate a caste system. Fit is shown through work and revisited, not assigned from a test.

8a.3 Why this protects the complexity principle

Two structural commitments follow, and they are what keep «authority by complexity» humane rather than hierarchical. First, **build roles must be protected from operative load**: if the initiators are buried in a run queue, build does not happen and the organization stops authoring itself. Second, **run roles must have a real path to authority and pay through complexity**, especially as AI raises the complexity of directing the run. Without the first, the organization

ossifies; without the second, «complexity-based authority» becomes a polite name for the old class system. The behavior chapter exists to hold both.

8a.4 The cultural failure modes

Turf behavior, public undermining, passive withdrawal, and, the one this edition adds, **status-by-headcount creeping back in**, where people compete to accumulate reports and budget rather than to hold harder problems. The symptom is a gap between the documented model and how daily life is actually described; the remedy is enforcement, visibly, through the bodies.

8a.5 A note on physical space

Where people sit is part of behavior, and it cuts both ways. A strong in-office culture is often assumed to be pure benefit, but for a HiveMind organization it carries a specific risk. Co-location manufactures exactly the kind of knowledge the framework works to avoid: decisions get made in spontaneous, unplanned, undocumented hallway conversations, and a great deal of important context ends up living only in the heads of whoever happened to be present. That is implicit knowledge by default, the opposite of structure before storage, and it quietly excludes everyone who was not in the room, including remote colleagues, future hires, and the AI participants that need a written trace to act. An office optimized for constant informal contact can therefore work against collective intelligence rather than for it.

The point is not that physical presence is bad. It is that presence should be spent on what it is uniquely good for. Routine coordination belongs in the shared environment, where it leaves a trace. In-person time is best reserved for the things text handles badly: long, well-prepared workshops on complex, ambiguous, or delicate questions, which is the high-dynamics build work of Chapter 5 and is far better

done in a room together. The output of those sessions is then written back into the environment, so the decision survives the meeting. Used this way, physical space serves the build mode while the run mode stays in the documented flow. Presence becomes a deliberate instrument rather than a default setting.

Part III

Implementation

The five pillars describe the destination. This part is the path. Chapter 9 is the staged playbook; Chapter 10 is the maturity model that locates an organization on it. Read through the run/build lens, the shape of the program is simple: **Year 1 is almost entirely build, and the run apparatus thickens only afterward.**

Chapter 9

The Implementation Playbook

Staged, because erecting all five pillars at once is a recipe for failure, and because the stages respect the dependencies. Three years is illustrative; the sequence matters more than the duration. The build-heavy front and the run-heavy tail are deliberate: you author the size-invariant layer once, then scale run onto it.

9.1 Preconditions

— **Leadership commitment** to the program as the primary operating-model change of the next three years, modeled from the top.

— **A diagnostic baseline** (the maturity model), one to four weeks, showing which pillars are present, absent, or lived-but-implicit.

— **A capability bridgehead.** Three to seven first movers who lead the program. Critics are bypassed, not converted; the bridgehead's success persuades.

9.2 Phase 1 — Doctrine & foundation (Year 1): the build year

Almost entirely *build*. Resist starting with tools.

— **1.1 Draft the doctrine** (six to twelve weeks, including the most senior leader; coached at most, not outsourced), including the

authority-by-complexity and compensation-by-complexity principles.

— **1.2 Define the cluster structure.** Names, scopes, exclusions, interfaces, candidate leads, and for each cluster the intended build line / run line split.

— **1.3 Inventory information resources.** The first catalog.

— **1.4 Choose the platform stack.** Only now, because doctrine and structure narrow the choice.

— **1.5 Formulate the AI policy.** Short, pragmatic, revised quarterly.

— **1.6 Train the bridgehead and leadership.**

> **Year 1 delivered:**
>
> ratified doctrine (v1.0); cluster map with leads, scopes, and build/run split; information inventory (v1.0); chosen stack; AI policy (v1.0); trained leadership. **Deliberately not yet:** org-wide migration, deep process documentation, AI beyond the bridgehead, restructuring beyond the cluster definitions.

9.3 Phase 2 — Architecture & adoption (Year 2)

Run begins to scale onto the built foundation. Migrate onto the platforms (cluster by cluster, information before communication, old systems archived read-only and shut off); document processes bottom-up (the longest and most valuable single activity); activate the cluster model and bodies; operationalize information governance; pilot AI augmentation with curation agents in low-risk domains.

9.4 Phase 3 — Augmentation & maturity (Year 3+)

Scale AI augmentation (analysis-level AI, composition designed per workflow, directing AI generalized); move from documented to living,

partly automated workflows; measure results; evolve the doctrine annually; iterate the maturity assessment.

9.5 What goes wrong with the program

Starting with tools; skipping the doctrine; a leadership exception (leaders who direct but do not adopt); half migration; process documentation as a one-off deliverable; AI before architecture; big-bang reorganization; and over-investing in run while never funding the build lines, the program version of the size failure from Chapter 1.

Chapter 10

The Maturity Model

10.1 Purpose

A diagnostic, not a grade. Five levels, derived from the eCH-0232 maturity model for information management and extended to all five pillars.

Level	Meaning
1 · Uncontrolled	Absent or ad-hoc. No shared rules, no documentation, no governance.
2 · Emerging	First rules and documentation in isolated areas; uneven.
3 · Forming	Established processes, named roles, documented standards; consistent.
4 · Operational	Fully operationalized: standards enforced, lifecycles governed, automation supports the work, metrics measured.
5 · Proactive	Continuous optimization; the pillar evolves deliberately on data and context.

Higher is not always better: a small organization may rightly sit at level 3. The model shows where you are; doctrine and strategy say where you should be.

10.2 Pillar I — Policy doctrine

Level	Characteristics
1 · Uncontrolled	No written doctrine. Principles are oral, contested, vary by leader. New decisions made from scratch.
2 · Emerging	Some written principles, often scattered (values, mission statements). No consolidated doctrine. References in decisions rare.
3 · Forming	A consolidated, ratified doctrine exists, is published and accessible. References appear in major decisions. Change process defined.
4 · Operational	The doctrine is actively used in operations. Decisions reference it routinely. Violations are addressed. Annual review. New members are inducted into it.
5 · Proactive	The doctrine evolves continuously from operational experience. Changes are openly discussed. It functions as a living constitution.

10.3 Pillar II — Cluster organization

Level	Characteristics
1 · Uncontrolled	Classic functional hierarchy. Authority flows upward. Decisions are escalated. Cross-functional work stalls.
2 · Emerging	Some teams with cluster-like traits (autonomy, a lead), but the model is partial and not formalized. The functional hierarchy remains primary.
3 · Forming	The cluster structure is defined, ratified, documented. Leads named, scopes and interfaces written, bodies with named participants and cadence defined.
4 · Operational	The cluster model is the actual structure. Decisions are made at cluster level. Bodies convene reliably. Behavioral expectations are enforced. The leadership body is a peer body. Authority and reward track complexity, not headcount.
5 · Proactive	The structure is deliberately reviewed and evolved. Scopes are adjusted, new clusters formed, old ones retired. Structure as evolving design.

10.4 Pillar III — Operational business architecture

Level	Characteristics
1 · Uncontrolled	Processes are implicit. Documentation, where present, is partial and outdated. New members learn by mouth.
2 · Emerging	Some processes documented (mostly the regulated ones). Uneven. The macro layer (value streams) does not exist.
3 · Forming	The macro layer is drawn. Major processes documented at meso level. Roles linked to processes. Documentation is consulted at least sometimes.
4 · Operational	The architecture exists at all three layers. Processes are lived, and work runs through the workflows. Automation supports steps. Continuous consultation.
5 · Proactive	The architecture is continuously refined. Process metrics measured, inefficiencies addressed. AI agents read the architecture and operate within it.

10.5 Pillar IV — Information governance

Level	Characteristics
1 · Uncontrolled	No formal governance. Classification ad hoc. Retention by individual habit. Access by chance knowledge. The environment is chaotic.
2 · Emerging	A classification scheme on paper, inconsistently applied. A partial inventory. Retention rules exist but are not enforced.
3 · Forming	Governance is defined: scheme, catalog over the essential resources, retention rules, access control, named owners. The catalog is consulted.
4 · Operational	Fully operational. The catalog is comprehensive and current. Classification at creation. Retention enforced. Access role-based and audited. Dev body convenes. AI policy integrated.
5 · Proactive	Governance evolves continuously. Quarterly audits, metrics on the health of the environment. Decisions increasingly automated under supervision. Regulatory answers without panic.

10.6 Pillar V — AI augmentation

Level	Characteristics
1 · Uncontrolled	AI is used by individuals on private accounts, without governance. The organization does not know which tools are used with which data.
2 · Emerging	An AI policy exists. Permitted tools listed, some prohibited ones flagged. Compliance uneven.
3 · Forming	The policy is operational. The classification-times-tool matrix is published and referenced. AI in certain workflows. Some workers practice the directing competence. Outputs with named accountability.
4 · Operational	AI augmentation is integrated into the architecture. Curation-level AI organization-wide. Analysis-level AI supports certain workflows. Compositions designed per workflow. Most knowledge workers direct AI.
5 · Proactive	The organization evolves its augmentation continuously. Compositions are reviewed. New capabilities introduced under the doctrine. It operates as a deliberate hybrid supermind.

10.7 Using the model, with the scale dimension

Baseline, set targets per pillar, reassess annually. Two additions in this edition:

— **The scale dimension.** Read every pillar's target against organization size. What «operational» looks like differs at **15 / 100 / 4000**: at 15, one person owning the catalog part-time *is* operational governance; at 4000, operational means an IG cluster with audits and tooling. The maturity level is the same word; its apparatus is not. A

useful worksheet adds a column, "what level 4 looks like at our size", per pillar.

— **Complexity-based authority and compensation are a level-4/5 capability.** Running authority and pay off complexity rather than headcount requires the lateral arbitration, the shared rubric, and the cultural trust that only exist at the upper levels. At levels 1–2 it is an aspiration in the doctrine; by level 4 it is enforced. Treat it as something the organization grows into, not a day-one switch, especially in regulated or pay-transparency environments, where it needs a defensible rubric.

10.8 A worked example: the typical small services firm

Pillar	Start	Target Y1	Target Y3
Doctrine	1	3	4
Cluster organization	1	3	4
Operational architecture	1	2	4
Information governance	1	2	4
AI augmentation	1	2	4

Doctrine and cluster organization lead (they are build, and cheap to author); architecture, governance, and AI follow (they are run, and scale with throughput). The pattern is the run/build logic expressed as a schedule.

Part IV

Placing the Framework

This part places HiveMind in its context: the macro frame of the network society, how the model extends across company boundaries, and the conclusion. The point-by-point distinction from related models now lives in Appendix F, so the through-line here stays positive. (The infrastructure layer, DSaaS, has likewise moved to Appendix E, where sovereignty is the actual subject, so the argument here stays universal.)

Chapter ii

Network Society, Network Organization

HiveMind is not merely a productivity program; it is the organizational form appropriate to a network society. Organizations built for the older form struggle in proportion to the gap between their structure and their environment.

11.1 The macro frame

The network society, a quarter-century old as theory and a decade old as operational reality, is defined by the primacy of flows: power, production, and culture organize around information flows rather than physical places; inclusion in the relevant networks is the condition of participation. An organization that keeps a non-network structure in a network society pays a permanent friction: its work happens through networks, its decisions through hierarchies, and the impedance mismatch burns energy. The five pillars are the structural correlates of network-society fitness.

11.2 Why large organizations stall: the run/build mechanism

This is the chapter where the run/build cut does its heaviest work, because it *mechanizes* a claim the white paper only asserted. Large organizations do not stall because their problems grow harder; in the typical case a 4000-person firm and a 100-person firm sell the *same* product at the *same* complexity. What grows is **run**: more people serving more clients with the same known product. The failure is to meet that growth by adding run on run, with more managers, more

coordination layers, and more headcount-based authority, while never funding the **build** layer that would give all that run its rails. Because build is size-invariant, the firm could have authored it once for roughly the same cost at any size; because it didn't, every new client adds coordination cost instead of riding a built rail. Friction compounds: a firm 10 percent slower per quarter than a network-fit competitor is ~50 percent behind within five years. The strongest form of the HiveMind argument is therefore not «you will be more productive» but «the alternative is structural decline as run outgrows an unbuilt foundation.»

11.3 The augmented network society

The original frame predates the algorithmic turn; networks now include algorithmic actors that generate content and make decisions. Two consequences: AI augmentation is not optional for a network-society organization (the environment's other actors are already augmented; the asymmetry is structural and growing), and governance becomes the central design question once algorithms participate in the network.

11.4 Implications for leadership

Leaders are network nodes, not throne-holders, and authority derives from position in the information flows and, per this book, from the complexity held, not the title or the headcount. Information is the lever (what a leader makes visible, amplifies, curates). The doctrine is the most consequential artifact of leadership. The leadership body is a body of peers serving the system, not a senior-management team the system serves.

11.5 The risk of the unfit organization

Compounding disadvantages in talent (network-native workers screen for operating-model fitness), speed (cluster-level vs. escalated decisions), AI integration (structure designed for AI vs. AI bolted on), and compliance, partnership, and resilience. None produces sudden collapse; each produces friction, and friction compounds.

Chapter 12

The Networked Organization: Working Across Company Boundaries

> **Build / run · scale**
>
> Deciding what the organization does itself and what it sources from partners is *build*, and it is among the highest-leverage build decisions a leader makes. Running the resulting relationships is *run*. The logic of the boundary is nearly size-invariant, but the line moves in practice as an organization grows: the larger it is, the more it can afford to hold in-house, and the more deliberately it must govern what it does not.

The network society does not stop at the edge of the firm. The same logic that makes a cluster the right internal unit, a bounded domain with a clear owner and clean interfaces, makes the single firm the wrong outer boundary for most work. An organization built on HiveMind principles is naturally porous: it keeps the capabilities that are core to its purpose and sources the rest from a network of specialized partners, joined by clear interfaces and clear contracts rather than by ownership.

12.1 The firm is not the natural unit of work

For most of the twentieth century the default answer to «should we own this capability?» was yes, because coordinating with an outside party was expensive and risky. That is the transaction-cost logic of Chapter 2: the boundary of the firm sits wherever it is cheaper to coordinate inside it than to buy across a market. When coordination

cost collapses, as it has, the efficient boundary moves inward, and many capabilities a firm once had to own can now be coordinated across organizations almost as cheaply as within one. The result is what the organizational literature has long called a network form, neither a pure market of arm's-length deals nor a single integrated hierarchy, but a durable web of specialized entities that coordinate through repeated dealing, reputation, and shared standards. Walter Powell set out this position as a third mode of coordination in 1990.

12.2 The conceptual API between organizations

What makes such a network work is the same thing that makes clusters work internally: a clear interface. Each entity is specialized, does its own thing well, and exposes a defined «conceptual API» to the others, a published account of what it provides, what it needs, and on what terms. This is the modularity argument from product design applied to organizations: a system can be split into independent modules as long as the interfaces between them are stable and well specified, and much of the value is captured at those interfaces, the «thin crossing points» between modules. The line of work runs from Herbert Simon's early account of near-decomposable systems through Carliss Baldwin and Kim Clark's design rules to the ecosystem research of Michael Jacobides and Annabelle Gawer. The mirroring observation of Chapter 2 carries straight across the boundary: the contract and the interface have to mirror the real dependency, or the relationship grinds. A clean interface between two firms is Conway's law honored on purpose.

12.3 What to hold and what to source

The practical question is which capabilities to keep and which to source, and a workable rule follows the run/build and complexity cuts of this book. Keep in-house what is high in complexity and close to the

organization's purpose, the work that defines it and changes fastest. Source what is well-defined, stable, and shared across many organizations, where a specialized provider has both deeper expertise and greater scale. Finance leadership, IT infrastructure, design, legal, and increasingly specialized analytical work are natural candidates, which is why fractional and shared-service arrangements such as a fractional CFO, a managed platform team, or an external design studio have spread so fast. Sourcing such work as a service rather than carrying it as permanent headcount is what lets an organization stay lean and agile: it pays for the complexity it actually needs, when it needs it, instead of building a standing function that then has to be kept busy.

12.4 Governing the network

A networked organization does not escape governance; it relocates it. Three things have to hold. The interface has to be explicit, with the conceptual API written down rather than assumed. The contract has to match the interface, so that the legal relationship and the working relationship describe the same thing. And the information governance of Chapter 7 has to extend across the boundary, because a partner that touches the organization's information is inside its classification scheme and its AI policy whether or not it is inside its walls. A shared platform often carries the coordination, with the same stigmergic mechanism that works inside the firm working between firms when several entities read and write a common environment under agreed rules.

12.5 The networked organization and the five pillars

This chapter does not add a sixth pillar. It observes that the five extend outward. The doctrine names which capabilities are core and sets the principles for partnering. The cluster model already treats each unit as a bounded domain with interfaces, so a partner is in effect a

cluster the organization does not own. The architecture documents the flows that cross the boundary. Information governance and AI policy travel with the information, not with the org chart. Read this way, the network of partners is the natural outer ring of the same operating system: clusters all the way out, joined by interfaces and contracts instead of by ownership.

Chapter 13

Conclusion

The thesis: organizations from 2026 onward should be «smart by design», collectively intelligent through the structure of their work, not the unsupported brilliance of their members. The argument rests on four claims, each defensible in its own literature:

1. **Intelligence is a property of structure** (collective intelligence): a well-designed organization of average people outperforms a poorly designed one of brilliant people. *(With the honest caveat of 1.2 on the reach of the evidence.)*
2. **The network society is the macro frame.** Organizations built for an earlier form face compounding structural friction.
3. **Hybrid human-AI groups are the new productive frontier.** Gains are real, but only when the integration is designed.
4. **The five-pillar synthesis is the operational answer.** Doctrine, clusters, architecture, governance, and AI, deployed together as one operating system, produce the network-fit organization.

Underneath all four runs the organizing idea this edition makes explicit: **complexity, not headcount, is what should organize a firm**: its authority, its compensation, and its very structure. Run scales with throughput; build is authored once; and the firms that thrive are those that get the build layer right and let complexity, not the size of anyone's team, decide who decides.

The architecture of the network society is being built this decade. The organizations that will operate effectively within it are forming now. The question is not whether the transition is made, but whether you lead it or follow it.

> ***The beehive is the metaphor; the supermind the goal; the doctrine the work.*** *The rest is implementation.*

Appendices

Appendix A

Policy-Doctrine Template

The following is a template structure for a HiveMind policy doctrine. The doctrine itself typically runs 15–25 pages. Sub-policies (information governance, AI use, records management) are separate documents referenced by the doctrine. The two principles new to this edition, authority and compensation track complexity, not headcount, live in sections 2 and 7.

Section	Content
1. Prime Directive	What this organization exists to do; what success looks like.
2. Fundamental principles	Information principles (Once-Only, Open by Default, structure before storage); decision principles (domain authority, lateral coordination, clarity over consensus); behavioral principles (ownership, proactive communication, direct challenge, no turf); augmentation principles (human accountability for AI, governed tool use); and the organizing principle that authority and compensation track complexity, not headcount.
3. Cluster structure	Cluster map; cluster definitions: scope, exclusions, interfaces (one section per cluster); coordination bodies (members, chair, cadence, scope); leadership body. Each cluster's intended build line and run line.
4. Decision rights	Decisions within a cluster; at a coordination body; at the leadership body; escalation and dispute resolution; how competing complexity claims are arbitrated laterally.
5. Information-governance position	Classification scheme (summary); retention principles; access and sharing principles; reference to the detailed governance policy.
6. AI-augmentation position	Permitted tools (summary); data-category-by-tool matrix; human accountability for AI outputs; reference to the detailed AI policy.
7. Behavioral expectations	What is expected of cluster leads; of all members; what is not tolerated; protection of build roles from operative load.
8. Review and change	Annual review process; change proposals; version control; communication to members.
9. Glossary	Definitions of key terms.

What the doctrine is, and is not

The doctrine is: a constitutional document, ratified, versioned, durable; operational (it produces answers in real decisions); falsifiable (its statements could be violated, and the violation would be visible); reviewed annually and changed deliberately. The doctrine is not: a policy manual (each sub-policy lives in its own document); a values statement (purely aspirational language is removed); a vision document (forward-looking ambition belongs in the strategy); a procedures manual (procedures live with the work, not in the doctrine).

Appendix B

Extended Maturity-Assessment Worksheet

This worksheet is the book's core diagnostic instrument. It extends the brief self-assessment from the maturity model (Chapter 10) into a complete, applicable tool. The author's guiding idea shapes its construction: it asks not «how good are we?» but «what does our evidence, above all our data catalog, say objectively about our maturity?» Maturity is not estimated but derived from demonstrable facts. For each pillar the worksheet provides: detailed assessment criteria across the five levels, an evidence-based control sheet to fill in, assessment questions, and an aggregation path to the pillar profile. This edition also adds an explicit **authority-by-complexity** criterion to Pillar II (B.2); the per-pillar **build/run and scale** guidance is carried by the chapter epigraphs and by §10.7's scale dimension, alongside which this worksheet is read.

B.0 How to use the worksheet

— **Step 1 — Gather evidence, not opinions.** For each criterion ask: which verifiable artifact proves fulfillment? An entry in the data catalog, a ratified document, a body's minutes, an audit log. If the artifact is missing, the criterion counts as unfulfilled, no matter how confident the participants feel.

— **Step 2 — Determine the level per criterion.** Using the criteria table, enter the applicable level (1–5) for each row. The level is the

highest one whose description is fully covered by evidence, not the one you are approaching.

— **Step 3 — Fill in the control sheet.** For each criterion, record evidence/data-catalog field, maturity level, status (✓ met / ◐ partial / ✗ open), gap with next step, and owner. The control sheet is the actual working document; the other tables are its guide.

— **Step 4 — Aggregate to pillar maturity.** Pillar maturity is conservatively the minimum of the load-bearing criteria (a pillar is only as mature as its weakest necessary criterion), supplemented by the average as a progress indicator. Rationale in B.7.

— **Step 5 — Transfer to the profile (B.8).** Current level, target level, gap, next step, and owner per pillar. The profile is what leadership reads.

> **The rating scale — status and maturity**
>
> Status per criterion: ✓ = fully supported by evidence · ◐ = partially supported, evidence incomplete · ✗ = unsupported or absent. Maturity per criterion (1–5): 1 Uncontrolled · 2 Emerging · 3 Forming · 4 Operational · 5 Proactive. A level counts as reached only when all lower levels are also fulfilled, so maturity is cumulative.

B.1 Pillar I — Policy doctrine

Assessment criteria across the five levels:

Existence & form: *1* No written doctrine; principles oral and contested. *2* Scattered principles, not consolidated. *3* Consolidated, ratified, published. *4* Actively used; new members inducted. *5* Living constitution, continuously evolved.

Referencing in decisions: *1* Never. *2* Rarely. *3* In major decisions. *4* Routinely in decision records. *5* References analyzed to sharpen the doctrine.

Enforcement of violations: *1* No concept of violation. *2* No consequence. *3* Noticed. *4* Reversed or trigger a change. *5* Violation patterns feed the evolution.

Change process: *1* None. *2* Informal. *3* Defined (annual review). *4* Consistently executed, versioned. *5* Proposals openly discussed, driven by operations.

Evidence-based control sheet — to fill in:

Criterion	Evidence / data-catalog field	Maturity	Status	Gap & next step	Owner
Ratified doctrine (date, version)					
Publication location / accessibility					
Doctrine references in decision records (last 12 months)					
Documented handling of violations					
Last annual review (date)					

Assessment questions. Is there a written doctrine? (No → 1) Ratified by leadership? (No → 2) Referenced in decisions? (No → 2)

Changed at least annually? (No → 3) Changes driven by operational experience? (Yes → 5)

B.2 Pillar II — Cluster organization

Named lead per domain: *1* Functional hierarchy. *2* Some teams cluster-like, not formalized. *3* Clusters defined, leads named, scopes written. *4* The cluster model is the actual structure. *5* Structure deliberately reviewed and evolved.

Real domain authority: *1* None. *2* Pseudo-autonomy, overrideable. *3* Scopes ratified. *4* Leads decide without approval within their domain. *5* Scopes adjusted as things change.

Lateral coordination (bodies): *1* Everything escalates. *2* Ad-hoc alignment. *3* Bodies with named members and cadence. *4* Bodies convene reliably and decide. *5* Body effectiveness measured and adjusted.

Authority by complexity, not headcount: *1* Authority follows headcount. *2* Implicit. *3* Principle written into the doctrine. *4* Complexity claims arbitrated laterally; reward decoupled from headcount in practice. *5* Continuously reviewed against a shared rubric.

Behavioral expectations lived: *1* Turf behavior normal. *2* Uneven. *3* Expectations written. *4* Visibly modeled and enforced by leads. *5* Part of hiring and development.

Evidence-based control sheet — to fill in:

Criterion	Evidence / data-catalog field	Maturity	Status	Gap & next step	Owner
Cluster map with named leads					
Scope/ exclusion/ interface definitions per cluster					
Body charter (members, chair, cadence)					
Body minutes from the last two cycles					
Example of a domain decision made without escalation					

Assessment questions. Is there a named cluster lead per domain? (No → level 1–2) Are scopes written and ratified? (No → level 2) Do leads decide without approval? (No → level 2) Do bodies convene on a defined cadence? (No → level 3) Is the structure deliberately reviewed? (Yes → level 5)

B.3 Pillar III — Operational business architecture

Macro layer (value streams): *1* Processes tacit. *2* Only regulated processes documented. *3* Macro map drawn. *4* Exists and is consulted. *5* Continuously refined with metrics.

Meso layer (processes per cluster): *1* None. *2* Uneven. *3* Main processes documented, roles linked. *4* Present at all levels. *5* Continuously improved.

«Lived» — work runs through the docs: *1* Outdated. *2* Rarely consulted. *3* Sometimes consulted. *4* Work runs through the flows. *5* Flows are the runtime; AI agents read and operate within them.

Engine room & automation: *1* None. *2* Isolated. *3* First automated steps. *4* Automation supports defined steps. *5* Process metrics drive optimization.

Evidence-based control sheet — to fill in:

Criterion	Evidence / data-catalog field	Maturity	Status	Gap & next step	Owner
One-page macro value-stream map					
Meso process docs per cluster (coverage)					
Role-to-process linkage					
Evidence that work runs through the flows (live example)					
Process metrics (cycle time, findability)					

Assessment questions. Is there a documented macro process map? (No → level 1–2) Are meso processes documented per cluster? (No → level 2) Is the documentation consulted in operations? (No → level 2) Does work run through the documented flows? (Yes → level 4) Are processes continuously refined against metrics? (Yes → level 5)

B.4 Pillar IV — Information governance

The most evidence-rich pillar and the heart of the fact-based logic: governance maturity can be read off the data catalog itself: its completeness, currency, and metadata depth are the objective measure.

Classification scheme: *1* None; ad hoc. *2* On paper, inconsistent. *3* Defined; most resources classified. *4* Applied at creation; drives access control. *5* Increasingly automated under supervision.

Data catalog (completeness): *1* None. *2* Partial. *3* Main resources captured. *4* Comprehensive and current (>80%). *5* Living; metrics on environment health.

Lifecycle & retention: *1* By habit. *2* Rules exist, not enforced. *3* Defined. *4* Enforced; deletion carried out. *5* Quarterly audits.

Access & permissions: *1* Chance knowledge. *2* Isolated restrictions. *3* Named owners, governed. *4* Role-based and audited. *5* Automated under supervision.

AI × classification integrated: *1* No connection. *2* Separate. *3* Matrix published. *4* Integrated into operations. *5* Continuously revised.

Evidence-based control sheet — to fill in (reference concrete data-catalog fields per row):

Criterion	Evidence / data-catalog field	Maturity	Status	Gap & next step	Owner
Classification scheme (levels, degree of application)					
Data catalog: share of resources with complete metadata					
Retention rules & evidence of executed deletion					
Permissions concept / role-based access matrix					
Last IG audit (date, findings)					
Classification × AI-tool matrix (version)					

Assessment questions. Is there a classification scheme? (No → level 1) Is there a data catalog? (No → level 2) Are retention rules enforced?

(No → level 3) Is the catalog comprehensive and current? (Yes → level 4) Are governance metrics tracked and acted on? (Yes → level 5)

B.5 Pillar V — AI augmentation

AI policy: *1* None. *2* Exists. *3* Operational; named accountability. *4* Integrated into the architecture. *5* Continuously adjusted.

Allow-list & tool matrix: *1* Unknown which tools used. *2* List exists, uneven compliance. *3* Matrix published. *4* Enforced. *5* New capabilities evaluated under the doctrine.

Curation before analysis: *1* None. *2* First curation agents. *3* Curation running, analysis piloted. *4* Curation widespread, analysis supports flows. *5* Hybrid supermind.

Human-AI composition designed: *1* Random. *2* Implicit. *3* Set in individual flows. *4* Designed per flow. *5* Periodically reviewed.

Directing competence: *1* None. *2* Isolated. *3* A few key people. *4* Most knowledge workers. *5* Generalized and developed.

Evidence-based control sheet — to fill in:

Criterion	Evidence / data-catalog field	Maturity	Status	Gap & next step	Owner
AI policy (version, accountability assignment)					
Allow-list / classification × tool matrix					
Curation agents deployed (domains)					
Analysis agents deployed (domains)					
Documented human-AI compositions per flow					
Share of knowledge workers with directing competence					

Assessment questions. Is there a written AI policy? (No → level 1) Is there an allow-list? (No → level 2) Is AI integrated into specific flows? (No → level 2–3) Are human-AI compositions designed per

flow? (Yes → level 4) Do most knowledge workers practice the directing competence? (Yes → level 5)

B.6 Initiative Check — the cross-pillar control sheet

Before any major change, such as a platform migration, a new AI deployment, or the founding of a cluster, an initiative check tests whether it honors the five pillars rather than undermining one.

Control point	Expected fulfillment	Status	Mat.	Remark / evidence	Owner
Doctrine coverage	Violates no principle; the decision right is clear.				
Cluster assignment	A named lead is accountable; interfaces are named.				
Process anchoring	Located in the architecture, not beside it.				
Information governance	Classification, retention, access governed; a catalog entry exists.				
AI conformity	Tools used are permitted; human accountability is named.				
Sovereignty / law	Data residency and subprocessor chains checked.				

B.7 From criteria to pillar maturity: the aggregation path

Maturity is not an average grade. A pillar whose load-bearing criteria stand partly at level 4 and partly at level 1 is not «level 2.5»; it has a critical gap the average would obscure. So two values are formed:

— **Load-bearing maturity (minimum).** The lowest level among the load-bearing criteria, those without which the pillar does not function (without a classification scheme, no information governance above level 1). This value goes into the profile.

— **Progress indicator (average).** The rounded mean of all criteria shows overall progress. When minimum and average diverge widely, that marks the gap to close next.

Criterion	Maturity	Reading
Classification scheme (load-bearing)	1	Caps the pillar.
Data catalog (load-bearing)	3	Devalued by the missing classification.
Lifecycle	3	
Access	2	
AI integration	2	
Load-bearing = 1	→	**Average = 2.2; the gap of 1.2 is the priority list.**

B.8 The pillar profile — the leadership view

Pillar	Cur.	Target	Gap	Next concrete step	Owner
Policy doctrine					
Cluster organization					
Operational architecture					
Information governance					
AI augmentation					

B.9 A worked example: a 100-person services firm

For illustration, a fictional mid-size services firm at the start of its HiveMind program. The assessment rests on what participants could objectively present, not on self-estimation.

Pillar	Cur.	Target Y1	Gap	Next concrete step	Owner
Policy doctrine	1	3	2	Draft doctrine v0.1 in 8 weeks (Prime Directive + 8 principles, incl. complexity-authority).	Exec
Cluster organization	1	3	2	Draw the cluster map; name leads; write scopes and the build/run split.	Exec
Operational architecture	1	2	1	Macro value-stream map; bottom-up docs in one pilot cluster.	IM
Information governance	1	2	1	Four-level classification scheme; data catalog v0.1 of main resources.	IM
AI augmentation	1	2	1	AI policy v0.1; allow-list; curation pilot (summarization).	IM

The firm starts at level 1 across all five pillars: no written doctrine, a de facto functional hierarchy, tacit processes, no classification, and AI on private accounts. This is the baseline, not a failure. Doctrine and clusters lead (target 3); architecture, governance, and AI follow (target 2). Every «next step» is tied to a verifiable artifact, so the next assessment again proceeds from evidence, not feeling.

The core idea of the worksheet

«Clarity creates trust.» A fact-based assessment takes the politics out of the maturity question: no one defends a grade, everyone reads the same evidence. The data catalog becomes the shared, objective mirror of maturity, and the gap between load-bearing maturity and average shows what to do next.

Appendix C

Glossary

Term	Definition
Augmentation	The extension of human work by AI participants designed as «new colleagues» in the supermind, not a bolted-on productivity gadget.
Build vs. run	Build is the work of authoring the organization (doctrine, cluster design, architecture); it is high-novelty, fast-changing, rare, and nearly size-invariant. Run is the work of executing the known; it is high-volume, stable, and scales with throughput.
Cluster	A bounded operational domain with a single named, accountable lead, defined scope, defined interfaces, and team membership.
Complexity (intricacy + dynamics)	The basis of authority and reward in this edition. Intricacy = how many interacting parts a decision has; dynamics = how fast the ground moves and how much of the work is novel. Authority width tracks complexity, not headcount.
Complexity-based authority and compensation	The principle that decision authority and pay follow the complexity a person handles, not the number of people they command; contestable and arbitrated laterally, never self-declared.
Curation before analysis	AI first orders the information environment (classify, tag, link) before being used analytically. Analysis without curation produces fast, plausible-sounding errors.
Data catalog / information catalog	The structured inventory of essential information resources with metadata, classification, lifecycle status, and ownership. Its maturity is the objective measure of information governance.
Doctrine (policy doctrine)	The constitutional layer, the kernel. A small, dense, written document defining Prime Directive, principles, decision rights, and the change process. Not a policy manual.

Term	Definition
DSaaS (Digital Sovereignty as a Service)	Companion offering to HiveMind. The infrastructure layer (the hardware): sovereign hosting, controlled SaaS retention, identity and access. See Appendix E.
Engine room	The most granular layer of the architecture: screens, keystrokes, API calls, and automated flows, where AI augmentation lives.
Human-AI composition	The deliberately designed mix of human and AI participation in a concrete flow. A design variable, not a discovered one; periodically reviewed.
Lateral coordination	Alignment between clusters through named bodies rather than escalation upward.
Leading store / single source of truth	The single canonical location where the current version of a resource lives. References link to it rather than copy it.
Maturity levels (1–5)	Uncontrolled, Emerging, Forming, Operational, Proactive. Cumulative: a level counts as reached only when all lower ones are fulfilled.
Once-Only	Information is captured once and reused, not re-entered.
Open by Default	Information is accessible to all who need it, unless a specific reason requires restriction.
Operating-system metaphor	The framing that binds the pillars into one machine: doctrine = kernel, clusters = processes-with-permissions, coordination bodies = scheduler, information governance = file system, AI agents = drivers/daemons, DSaaS = hardware.
Prime Directive	The one-sentence mandate of the organization: what it exists to do.
Stigmergy	

Term	Definition
	Coordination in which actors leave traces in a shared environment that others respond to, without direct command. The shared platform takes on this role.
Structure before storage	Information is structured at creation, not retrofitted later.
Supermind	A group, whether people, machines, or both, that acts collectively in a way that appears intelligent; a term from research on collective intelligence.
Tacit-to-explicit (knowledge spiral)	The conversion of tacit, in-the-head practice into explicit, shareable structure; the basis of the bottom-up «document your own work» method.
The two modes (initiating / sustaining)	The two kinds of attention work demands: initiating (build) needs slack and divergent attention; sustaining (run) needs flow and consistency. People are matched to modes by demonstrated strength and preference, reversibly.
Transactive memory	The formal account of «knowledge lives in the environment, not in heads»: who-knows-what (and where-it-lives) becomes a property of the group.
Clarity creates trust	The guiding maxim: explicit rules and visible evidence take the politics out of decisions and create the liberating power of clear rules.

Appendix D

The Five Pillars as a System

A closing synthesis: how the five pillars work together in operation. They are not independent modules; each presupposes and supports the others. This dependency table complements the operating-system mapping in Chapter 3 (kernel / processes / scheduler / file system / drivers / hardware): the OS view shows what each pillar *is*; this view shows how they *transmit force* to one another.

From	To	Relationship
Doctrine (I)	Cluster (II)	Cluster definitions are derived from the doctrine's principles, which scopes, leads, and bodies reference.
Cluster (II)	Architecture (III)	Processes are documented within clusters; interfaces defined at the cluster boundaries.
Architecture (III)	Info Gov (IV)	The architecture describes the flows; governance steers what flows. Each needs the other.
All (I–IV)	AI (V)	Agents operate on the architecture (what to do), within the governance regime (which data), under cluster authority (reporting to whom), in line with the doctrine (which principles).
AI (V)	Supermind	Amplifies the whole: a coordinated mesh of human and algorithmic intelligence that produces capability no component could produce alone.

When all five pillars are operational, the organization is a hybrid supermind: under shared principles, with explicit accountability, in a structured information environment. That is the goal. The path to it is the work of the transformation program, roughly three years. The reward is structural fitness for the network society of the late 2020s and 2030s; the alternative is structural resistance that compounds over the same period.

> *The beehive is the metaphor. The supermind is the goal. The doctrine is the work. The rest is implementation.*

Appendix E

Relationship to Digital Sovereignty as a Service (DSaaS)

Moved here from the body so the main argument stays universal; this is the infrastructure layer, where the Swiss sovereignty material properly lives.

HiveMind and DSaaS are companion offerings at different layers of one stack: HiveMind is the operating-model layer, DSaaS the infrastructure layer beneath it (the *hardware* in the operating-system metaphor).

The two-layer stack. HiveMind is doctrine, clusters, architecture, governance, and AI («how the organization works»). DSaaS is sovereign hosting, identity and access, self-hosted alternatives where needed, controlled retention of US-based SaaS, the technical base of information governance («the substrate it runs on»). Separable (each addresses what the other does not) but connected (each presupposes the other).

What DSaaS addresses (Swiss context). Data residency (storage and processing in Switzerland, under Swiss law); sovereign hosting alternatives where US-hosted SaaS brings CLOUD-Act exposure (Nextcloud, Vaultwarden, Outline, ONLYOFFICE/Collabora, own mail); controlled retention of US-based SaaS where business value justifies it, with explicit data-category rules (the classic case being a cloud-based project-management tool); identity, access, and zero trust; and AI-tool governance at the infrastructure level (allow-lists enforced through network and platform controls, subprocessor jurisdictions checked, in-tool AI features explicitly governed).

Why two products. Some organizations need DSaaS but not yet HiveMind (an acute sovereignty trigger, an operating model they do not yet want to change); some need HiveMind but not yet DSaaS (competent infrastructure, an operating model that has not kept pace); many need both, sequenced.

Sequencing. *A, DSaaS first,* where sovereignty pressure is acute and the current model can survive a year or two on new infrastructure. *B, HiveMind first,* where the operating-model pressure is acute and sovereignty is not yet operational. *C, in parallel,* for greenfield or strong-leadership situations.

> **The author's preference**
>
> A or C. Sequence B carries a hidden risk: building HiveMind on insufficient infrastructure reaches the limits of its sovereignty posture exactly when the AI-augmentation phase (Year 2–3) demands more of the infrastructure, so the infrastructure pressure arrives as a crisis rather than a planned transition.

Appendix F

Anti-Patterns: What HiveMind Is Not, and Why That Matters

HiveMind is occasionally read as a variant of models from which it deliberately differs, and the distinctions reflect specific lessons from those models' failures. This appendix gathers the «what the framework is not» material, so it is said once, in one place.

F.1 Its nearest honest cousin: Team Topologies

The fair comparison, the one the white paper missed by reaching for Holacracy, is **Team Topologies.** Its four team types (stream-aligned, platform, enabling, complicated-subsystem) and its explicit interaction modes are close kin to HiveMind's clusters, its IG and architecture functions, and its build lines. Where HiveMind adds beyond it: the constitutional doctrine above the teams, information governance and AI augmentation as first-class pillars, and the complexity-based authority and pay logic. Team Topologies is a strong account of how to shape teams and their interactions; HiveMind embeds that account inside a full operating system.

F.2 Not Holacracy

It shares some DNA with Holacracy: a written constitution, bounded domains, and a rejection of manager-subordinate as the primary axis. The essential differences are that HiveMind is process-light where Holacracy is process-heavy, names accountable *people* per cluster where Holacracy distributes authority by role (person-bound

accountability resists dilution), and aims at *designed* coordination rather than emergent self-organization. Holacracy's documented failure modes, including no clear cross-circle authority, governance overhead, and hard strategic implementation, are things HiveMind's named-lead and lateral-body design addresses directly.

F.3 Not Teal

It shares Teal's rejection of pure command-and-control, but Teal de-emphasizes structural mechanics where HiveMind is explicit, aims at consensus where HiveMind chooses clarity over consensus, and leaves purpose to emerge where HiveMind writes it down.

F.4 Not «agile transformation»

It overlaps with agile on small teams and increments, but agile targets software delivery where HiveMind targets the operating model, is structure-agnostic where HiveMind is structure-specific, and is silent on information governance and AI where HiveMind makes both pillars. Agile is a practice inside clusters, not a substitute for the cluster-and-doctrine structure above them.

F.5 Not «roll out a tool» and not «adopt AI»

These are the two most expensive confusions. A tool is necessary and about a tenth of the work; without doctrine, structure, architecture, governance, and policy it is just another archive. And AI placed on an existing operating model yields local efficiency, not transformation, which is exactly why AI is the fifth pillar and not the first.

F.6 What this appendix comes down to

HiveMind sits at a specific place: more explicit than Teal, more process-light than Holacracy, broader than agile, deeper than a tool rollout or an AI rollout, and one constitutional layer above Team Topologies. What it rejects in each case is the *sufficient* version of the alternative. The sufficient answer is the five-pillar synthesis, organized as one operating system.

Bibliography

The works on which this book rests, arranged by theme. The text names no names in its body, so the arguments stand on their own; here is what the synthesis owes. This edition expands the list to credit the ancestors named in Chapter 2.

Network Society and Social Theory

— Castells, Manuel. *The Rise of the Network Society. The Information Age, Volume I.* 1996.

— Castells, Manuel. *Communication Power.* 2009.

Collective Intelligence and Superminds

— Malone, Thomas W. *Superminds: The Surprising Power of People and Computers Thinking Together.* 2018.

— Malone, Thomas W., and Michael S. Bernstein (eds.). *Handbook of Collective Intelligence.* 2015.

— Woolley, Anita Williams, et al. *"Evidence for a Collective Intelligence Factor in the Performance of Human Groups." Science 330.* 2010.

— Zhang, Shaowu, et al. *"Honeybee Memory: A Honeybee Knows What to Do and When." Journal of Experimental Biology 209.* 2006.

— von Frisch, Karl. *The Dance Language and Orientation of Bees.* 1967.

Hybrid Human-AI Intelligence and its Forebears

— Vaccaro, Michelle, Abdullah Almaatouq, and Thomas W. Malone. *"When Combinations of Humans and AI Are Useful: A Systematic Review and Meta-Analysis." Nature Human Behaviour.* 2024.

— Cui, Hao, and Taha Yasseri. *"AI-Enhanced Collective Intelligence." Patterns.* 2024.

— Licklider, J. C. R. *"Man-Computer Symbiosis." IRE Transactions on Human Factors in Electronics.* 1960.

— Engelbart, Douglas C. *"Augmenting Human Intellect: A Conceptual Framework." SRI.* 1962.

Governance, Architecture, and the Economics of Organization

— Ostrom, Elinor. *Governing the Commons: The Evolution of Institutions for Collective Action.* 1990.

— Beer, Stafford. *Brain of the Firm* (the Viable System Model). 1972.

— Galbraith, Jay R. *Designing Complex Organizations* (the information-processing view). 1973.

— Conway, Melvin E. *"How Do Committees Invent?"* (Conway's Law). 1968.

— Coase, Ronald H. *"The Nature of the Firm." Economica.* 1937.

— Williamson, Oliver E. *The Economic Institutions of Capitalism* (transaction-cost economics). 1985.

Networks, Modularity, and the Boundary of the Firm

— Powell, Walter W. *"Neither Market nor Hierarchy: Network Forms of Organization." Research in Organizational Behavior 12.* 1990.

— Baldwin, Carliss Y., and Kim B. Clark. *Design Rules, Volume 1: The Power of Modularity.* 2000.

— Jacobides, Michael G., Carmelo Cennamo, and Annabelle Gawer. *"Towards a Theory of Ecosystems." Strategic Management Journal 39.* 2018.

Organization and Operating Models

— Skelton, Matthew, and Manuel Pais. *Team Topologies: Organizing Business and Technology Teams for Fast Flow.* 2019.

— Bersin, Josh. *"The New Organization: Different by Design." Deloitte Global Human Capital Trends.* 2016.

— Laloux, Frederic. *Reinventing Organizations.* 2014.

— Robertson, Brian J. *Holacracy: The New Management System for a Rapidly Changing World.* 2015.

— Snowden, David J., and Mary E. Boone. *"A Leader's Framework for Decision Making" (Cynefin). Harvard Business Review.* 2007.

— McKinsey & Company. *"Unleashing the Power of Agile Teams" and "Organizing for the Age of Urgency." Reports on organizational design.* 2018-2023.

Knowledge Mechanisms

— Nonaka, Ikujiro, and Hirotaka Takeuchi. *The Knowledge-Creating Company* (the tacit-to-explicit / SECI spiral). 1995.

— Wegner, Daniel M. *"Transactive Memory: A Contemporary Analysis of the Group Mind."* 1987.

— Grassé, Pierre-Paul (origin of *stigmergy*); Heylighen, Francis (stigmergic human coordination).