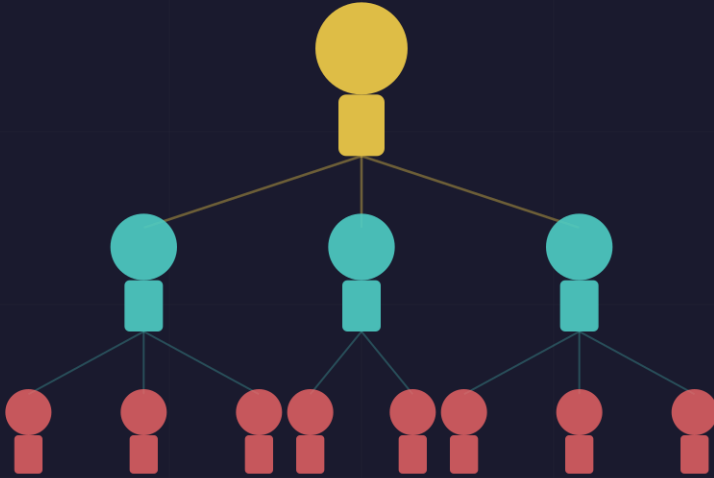


PRELIMINARY COVER



Organisational Dysfunctions

Open Systems Theory explains

Trond Hjorteland

Organisational Dysfunctions

Open Systems Theory Explains

Trond Hjorteland

This book is available at <https://leanpub.com/organisationaldysfunctions>

This version was published on 2026-07-03



This is a [Leanpub](#) book. Leanpub empowers authors and publishers with the Lean Publishing process. [Lean Publishing](#) is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

© 2026 Trond Hjorteland

For Merrelyn, who had the nous to see what the software industry could become, and the generosity to help someone try.

Contents

Foreword by Merrelyn Emery	1
Foreword by Alexandra Stokes	2
Preface	3
 What the Structure Does	 9
What the coal miners discovered	10
The two design principles	11
Modern forms of laissez-faire: the third state	14
Why structure beats culture	15
The six psychological requirements	15
 In the Room	 16
The room that went quiet	16
The phones came out	16
The pair that runs everything	16
The agenda that sabotaged itself	16
The output nobody owned	17
HiPPOs and dungeon masters	17
Tyranny of the majority	17
The corridor conversation	17
 In the Team	 18
The daily status report	18
Passing the buck	18
The powerless retrospective	18
Forming–storming–norming–performing	18
The hero culture	18
The servant who still decides	18

CONTENTS

Working alone together	19
Out of sight, out of sync	19
The agile terrarium	19
The collaboration that isn't	19
The product owner trap	19
Analysis paralysis	20
The decision that went nowhere	20
Permanent urgency	20
Rearranging the furniture	20
In the Department	22
You are empowered now	22
Psychological safety as a patch	22
DORA, the wrong way round	23
Team Topologies, the wrong way round	23
Involvement theatre	23
Better leaders is the answer	23
Change agents of the status quo	23
The external verdict	23
The frozen middle	24
It's a communication problem	24
Them and us	24
Fixing people	24
The error factory	24
Burned by design	24
Designed to undermine	24
The pilot trap	25
Fixing the process	25
Local optimisations	25
In the Organisation	26
OKRs imposed from above	26
Budgets are bureaucracy	26
The performance review	26
The learning organisation that doesn't learn	27
The short-termism machine	27
The strategy that did not survive contact with reality	27
The leadership team that isn't	27
Quiet quitting	27
The Sunday email	27

CONTENTS

It's just a job	28
The project in product clothing	28
Pay and reward	28
Career paths	28
The agile scaling trap	28
Intermezzo: Objections	29
Self-management doesn't scale	29
Nobody actually does this	29
All teams need a manager	29
This is fifty-year-old theory	30
This sounds like communism	30
IT is different	30
Agile works fine, why do we need OST?	30
Developers just want to code	30
We've already transformed	31
What the Open System Does	32
Where the world is now	32
Active and passive adaptation	32
Beyond the Walls	33
Built for yesterday	33
Deploying AI into a broken system	33
The customer we never met	33
The AI we cannot talk about	33
The market we think we shape	33
Outsourcing the future	33
People resist change	34
Doing the wrong thing right	34
The Way Out	36
The Two-Stage Model for Structural Change	37
The Search Conference	37
The Participative Design Workshop	37
Epilogue: The Diagnosis	38

Glossary	39
Acknowledgements	40
About the Author	41

Foreword by Merrelyn Emery

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Foreword by Alexandra Stokes

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Preface

Some years ago, I was the solution architect on a brownfield project. A new strategically important product was to be built, which required a large rewrite of the old existing monolithic application, and, consequently, the surrounding services had to be modified. Not only because of these new additions, but mostly due to adjustments to the core and its APIs. A massive undertaking that was too complex for a single person to keep in their head. Especially me.

I had been brought in for my experience, and it was exactly that experience that told me clearly that designing this thing alone was a fool's errand. So I did something I had learned from the Domain-Driven Design (DDD) community; I told the team we were going to design it together. I had no intention of being the master designer; all architecturally relevant decisions should be done collaboratively. I was one of them, with the perspective of someone who had seen a few of these complex undertakings before, but they were the experts on this product, this codebase, and this domain. We had to model together, decide together, and own the design together. Simply put, I was not equipped to do this alone, and I knew, without telling them, that they would probably never accept my design regardless of how perfect it would be. Which it would not.

It worked, and not in a small way. It worked the way the Agile Manifesto promises but rarely delivers. The team came alive. The architecture they shaped collectively was better than anything I could have dreamt of drawing on my own. The line management saw it, the project lead saw it, and the fellow coders in the department were impressed. I felt I really had made a proper change for once, found a way out of the frustration of being a foreigner in their system and being expected to perform miracles. Lifted tons off my shoulders. And brought excitement to us all. My imposter syndrome was nowhere to be seen, even.

Then the estimates began to slip, and we started missing expected deadlines. We had been clear from the start that the estimates were pure guesses, as the rewrite was something none of us had done before, and the monolith had so much technical debt and history that was older than all of us. The system landscape was complex enough that no one could really know how long this would take, and we had said so, in writing, more than once. But

complexity is uncomfortable for organisations that need to plan, and a few months in, the planning anxiety started to outweigh the evidence. A new layer of management arrived as well, who had not been part of the work up until then and did not know the team or the systems. They looked at the slipping numbers, looked at the architect with the title and the grey beard, and made the decision they thought was responsible. They asked me to take control.

I tried to explain that the team had earned the design, that taking control from them would dismantle the thing that was making the work possible. It did not land, and the pressure mounted. The trust the team and I had built collapsed in a matter of weeks, replaced by the older and more familiar arrangement where one person directs and the others execute. The team even accepted it as they were all too used to the idea that this is how things really work, and that happy bubble we had was just that, a fleeting and fragile soap bubble.

I left not long after. I was disillusioned in a way I had not been before, and I did not yet have the language nor the understanding of what had happened. I think I know now and would like to share it with you all. Hoping that it can help you avoid the hit that we all took back then.

* * *

What I had seen was real; the work had been good, and the collapse had unfortunately been equally real. Something had been working, and then something else had broken it, and I could feel the shape of both even though I could not yet name them. So I began to read. A lot.

My route in was systems thinking. I had been drawn to it for a while, partly through the DDD community and a colleague who could not stop talking about *Thinking in Systems* by Donella Meadows. I had also been an avid follower of Ruth Malan's work on architecture and scopes, where there are different levels of architectural decisions to be made, ranging from the details in the code to large-scale enterprise-level decisions. That, and some really inspirational recordings of Russell Ackoff on YouTube, gave me a vocabulary for why some messy problems do not yield to being broken into neat little parts. Some of the material also kept pointing back to something older and stranger, something called sociotechnical systems design. I followed the thread, that rabbit hole.

And that was a deep one. It led me to the Tavistock Institute of Human Relations and to the British coal mines in the 1950s, where researchers

had found that workers organising themselves into self-managing groups produced better results than the supervised division of labour the new technology had imposed. It led to a chapter of Norwegian industrial history I knew, embarrassingly enough, nothing about. In the 1960s, as part of the Industrial Democracy Programme, Fred Emery and Einar Thorsrud ran field experiments across four companies on whether workplaces could be made genuinely democratic. It then led me to Fred Emery's later work, the Open Systems Theory, which I would come to rely heavily on, where the design of organisations is treated as a question to be answered by the people who do the work, rather than by external experts. Felt emotionally familiar.

What struck me most was a single insight that ran through all of it. It is only when people design their own work that they develop the motivation, responsibility, and commitment to implement it well. Fred and his partner Merrelyn Emery had written this back in 1974,¹ and by pure coincidence, I had stumbled onto a version of it through collaborative modelling in the DDD community half a century later. The miners had lived it in the 40s and found the same solution as we did in that monolith rewrite. I had assumed I was doing something new on that brownfield project. I was doing something very old and deeply human, it turned out, but unfortunately, with most of the supporting structure missing. Which led to its unavoidable demise.

The next thing that struck me was the language of organisational design principles. Emery had named two basic structural patterns on which organisations can be built. The first is where coordination and control live above the work, in supervisors, managers, and architects. The second is where coordination and control live with the group doing the work. He had shown, with research, even in my Norway, that these two cannot be mixed.

The attempt to mix them actually resulted in a mixed mode with no clear home for coordination and control. The Emerys clearly showed that this adds confusion to the mix, as nobody really knows who calls the shots, and it results in what they called *laissez-faire*, which performs worse than either of the other two. Even worse than the slow and rigid bureaucracy. That was such an a-ha moment for me. The brownfield project had been an island of the second principle inside an organisation built on the first. The collapse had been the first principle reasserting itself under pressure, which is what it does when its assumptions are challenged. The vocabulary made what I had

¹Emery, F. E., & Emery, M. (1974). *Participative design: Work and community life*. Centre for Continuing Education, Australian National University. Available at socialsciencethatactuallyworks.com.

lived through legible.

I have spent the years since trying to bring this thinking into the IT industry. The research behind it spans sixty years and dozens of industries: coal mines, chemical plants, shoe factories, oil refineries, wineries, hospitals. The results are consistent and well-documented. The IT industry is conspicuously absent from that list. This book is a modest attempt to change that. The industry has its own version of the same story. It has been trying to move toward something like self-management for over two decades under the banner of Agile, and has produced mostly laissez-faire instead. The research is precise on this,² and the experience confirms it. Most of us in the industry have lived some version of what I lived through. Most of us do not have the language for it.

This book is an attempt to give us that language and show you where and when to recognise it.

The examples in this book are from software organisations because that is where I have spent my working life. But the structural patterns are not industry-specific. Wherever people come together with a shared purpose, whether at work, in a community, an NGO, a volunteer project, or a neighbourhood committee, the same design principles apply, and the same dysfunctions follow from the same structural causes. The coal miners knew this. The Norwegian factory workers knew this. The IT industry is simply the latest to rediscover it.

* * *

This book grew out of a daily LinkedIn series. The format it found there, one dysfunction at a time, named the way people experience it, turned out to be the right one.

This book is therefore mainly a catalogue of organisational dysfunctions, each one named the way people experience it rather than the way theory describes it. Power asymmetry, not communication problems. Powerless retrospective, not ignorant developers. The incapacitated middle managers, not an agile transition in progress. The unrelatable mission statement, not people resisting alignment. Each entry sets up a situation that anyone

²Merrelyn Emery, “A Patchwork of Contradictions and Confusions: Inside the Software Industry,” *Social Science That Actually Works*, 2023, socialsciencethatactuallyworks.com.

who has worked in a software organisation will recognise immediately, then explains it through the lens of Open Systems Theory.

The aim is twofold. The first is recognition. If you have been carrying a dysfunction around without language for it, this book may help you name what you are inside of. Naming is not the same as fixing, but it is the precondition for fixing. You cannot redesign a system you cannot describe.

The second aim is structural. Most of the dysfunctions in this book are usually explained as failures of culture, leadership, communication, or individual character. Almost none of them are. They are structural outputs of a particular way of organising work, the bureaucratic hierarchy that has been the default for most of the industrial era and most of the IT industry within it. Once you see this, you stop trying to fix the people inside the structure and start asking different questions about the structure itself. We need to get out of this bureaucratic hangover we seem to be still in, Agile or not.

This book does not, however, give you a method for changing your structure. That will be the work of a different book, which I am writing with expert facilitator and organisational designer João Rosa, called *Intentional Organizations*, which will go deeper into the practice of structural change. What I do here is name what is broken, with enough precision that the work of redesign becomes thinkable. Diagnosis before treatment. Most organisations skip the first step and then wonder why the second step does not take.

* * *

A note on how to read this. The book is organised into five chapters that move from the most intimate dysfunctions to the most systemic. Chapter one is about workshops and meetings, where dysfunction is most visible in the moment. Chapter two is about teams; Chapter three is about departments and the middle layers of organisations; Chapter four is about the organisation as a whole; and Chapter five steps outside and looks at the organisation in its environment, where Open Systems Theory has its sharpest things to say.

There is also an intermezzo, midway through, where I take on the objections that anyone serious about this work eventually hears. That self-management does not scale; teams must have leaders; it will never work for knowledge work like IT; and why bother, Agile works fine. None of these survives contact with the evidence, but they are honest objections and deserve honest answers.

You can read the book straight through, which I recommend if you are new to the framework. You can also dip in. Each entry is built to stand alone, and the section introductions repeat enough of the vocabulary that you can land anywhere without losing your footing. The book is also designed to be read slowly. One entry a day is the original cadence of the series, and I still think it is the right cadence. Each diagnosis takes a moment to sink in. There is no rush. Download it to your favourite e-reader and read one entry on the commute to the office, and then compare it to what you observe there.

There is a third way. The part introductions, the intermezzo, and the two closing chapters together form a coherent, standalone introduction to Open Systems Theory, readable in an evening without visiting any of the individual entries. If you want the framework before the catalogue, or if you come back to the book later and want a refresher without the full catalogue, that skeleton works on its own.

A note on versions. This book will evolve as good digital books ought to. The LinkedIn series on [linkedin.com/in/trondhjort](https://www.linkedin.com/in/trondhjort) continues to publish new entries, and each new release of the book incorporates them. The version you are reading is v1.0: a complete and coherent argument, but not the final word. Later releases will add entries to every chapter and deepen the intermezzo. If you are reading this on Leanpub, you will be notified when a new version is available.

A final word before the book begins. The dysfunctions in this catalogue are real. I know. They are also painful, particularly for people who have spent years inside them without knowing what they were inside of. If you find yourself recognising too much, that is not a sign that something is wrong with you. It is a sign that the structure you are working in is producing what it is designed to produce. Most of us have been there. Most of us are still there. The first step out is being able to name where you are.

What the Structure Does

“The product of work is people, as well as goods and services. A society is no better than the quality of the people it produces.”

– David (P.G.) Herbst¹

Herbst’s claim is not sentimental. It is a design statement: if the product of work is also people, then how work is organised is not just an efficiency question. It is a question about what kind of people the organisation is producing. To answer that well, you need more than experience. You need a framework that reliably tells you why some arrangements produce flourishing and others produce harm, regardless of the people within them. In this book, that framework is Emery’s Open Systems Theory.

It clearly explains that the dysfunctions in the chapters that follow are not just random irritations, and they are not failures of individual character. They are outputs of structure. Different ways of organising work produce different families of dysfunction with a predictability that, once you see it, is hard to unsee.

This introduction sets the stage. It starts with where it comes from, because the research is older than most readers will expect, and that history matters. It then introduces the two structural patterns that organisations can be built on, and the third pattern that appears when those two are mixed or muddled. As Wengrow and Graeber show in *The Dawn of Everything*,² humanity somehow decided that one of those patterns is the default when scaling to large social groups, a pattern solidified at work by the mechanisation of the industrial revolution. So much so that very few see the other alternatives,

¹David (P.G.) Herbst, *Socio-technical Design: Strategies in Multidisciplinary Research* (London: Tavistock Publications, 1974).

²David Graeber and David Wengrow, *The Dawn of Everything: A New History of Humanity* (London: Allen Lane, 2021).

even when the people working there do their damndest to change that, which, unfortunately, makes things even worse, as we will see.

We will also describe the best tool to judge which structural pattern works or not: the six conditions that determine whether the people working inside such a structure are likely to flourish or wilt. Part Two will take the same framework outside the walls. For now, we stay inside.

What the coal miners discovered

In the late 1940s, researchers from the Tavistock Institute for Human Relations in London were invited into the British coal mines. The mining industry was modernising by introducing mechanical coal-cutting equipment that promised great efficiency improvements. With it came the supervisory and command-and-control management style, which had become standard in the industries of the day. Treating labour as an engineering problem, with division of labour and narrow tasks. The work was organised into distinct shifts that did highly specialised work in phases, where one would hand over their results to the next. The arrangement everyone now takes for granted as to how work should be organised. Taylorism by the book. People are reduced to an extension of the machine rather than the tool in their hands.

Surprisingly, it was producing terrible results. Productivity was poor, accidents were frequent, absenteeism was high, and morale was awful. Something was clearly wrong, but the assumption was that more of the same medicine would cure it: better training, stricter discipline, smarter supervisors, tighter control.

What the Tavistock researchers found was that some coalfaces had quietly gone back to an older way of working. Miners had reorganised themselves into self-managing groups that shared the whole task between them, rotated roles, coordinated their own work across the shift, and held themselves accountable for their results. The supervisors above them, literally on the surface in this case, had been left in place formally, but no longer ran the work in practice. These coalfaces were outperforming the others by considerable margins, with better safety records and higher worker satisfaction.

This is where sociotechnical systems design begins. The researchers, Eric Trist among them, recognised that something fundamental was being demonstrated. Organisations are simultaneously social and technical systems, and the two must be jointly optimised: fitting together the

best version of the social structure with the best of the technology. The technology of the new coal-cutting machinery was almost identical in both arrangements. What differed was the social structure around it, and that difference was decisive.

The insight has been confirmed repeatedly over the decades, not particular to coal mining. In the 1960s, Einar Thorsrud, joined by Fred Emery from the Tavistock Institute, ran field experiments in the Norwegian Industrial Democracy Programme across four sites: wire drawing at Christiania Spigerverk, paper production at Hunsfos Fabrikker, panel heater manufacturing at NOBØ, and fertiliser production at Norsk Hydro. All four demonstrated the same effects. Norwegian working life owes a great deal to this work. It is a pillar of the Nordic work model, and the findings are encoded directly in the Work Environment Act: section 4-2 requires that employees are given opportunities for self-determination, influence, professional responsibility, variation, and the chance to see the connection between individual tasks and the whole. The research has since continued in healthcare, manufacturing, knowledge work, and increasingly in software. The conclusions have not changed.

The agile manifesto came along five decades after the coal mines and reached, in some respects, similar conclusions: small teams, self-organisation, and direct collaboration. The agile movement did not know it was reinventing sociotechnical systems design, and its lack of grounding in the design principles has cost it dearly, but the parallels are there for anyone who looks. Most of what is broken in modern software organisations was named, explained, and partly solved fifty to seventy years ago. We keep relearning the same lessons because we keep applying the same structural pattern. This book is an attempt to make that pattern visible.

The two design principles

In 1967, during the Norwegian programme, Fred Emery identified what he called the two genotypical organisational design principles. The word “genotypical” matters. He borrowed it from genetics to make a particular argument. The two patterns he was naming are not styles, preferences, or cultural variations. They are structural patterns at the deepest level, in the same way that genotypes underlie the surface variety of organisms. The superficial variations of most large organisations are phenotypes. The underlying genotypical design principle is the same. This is why most restructuring

in large companies makes little lasting difference: the phenotype changes, the genotype does not.

Emery called them DP1 and DP2; Design Principle 1 and Design Principle 2. At their core, they represent a single structural choice: where do you place responsibility for coordination and control? Over the work, or with the people doing it? What follows is a precise definition lifted from the OST literature, where every word matters.

Design Principle 1: The first (DP1) is called *redundancy of parts* because there are more parts (people) than are required to do the work at any one given time. Its critical feature is that responsibility for coordination, the horizontal dimension of organisational structure, and control, the vertical dimension, is located at least one level above where the work or planning is being done. Its basic module is a small section consisting of some people all doing small, different parts of a whole task with a supervisor. It is the supervisor who holds the responsibility for the coordination and control of all the functions that cover, such as goal setting. Therefore, the DP1 organisation is autocratic or bureaucratic, relying on the master-servant relationship in action, where those above have the right and responsibility to tell those below what to do and how to do it. It is a structure of personal dominance, or a dominant hierarchy. People are of unequal status.

Design Principle 2: The second organisational design principle (DP2) is called *redundancy of functions* because the more skills and functions are built into every person, the more that person can use at any point in time. Its critical feature is that responsibility for coordination and control is located with the people doing the work or planning. The basic module here is a self-managing group that works collectively to complete the whole task of the section. It is the group that sets the comprehensive set of measurable and agreed goals. DP2 structures are participative democracy in action, and large DP2 structures are non-dominant hierarchies of function in which all change is negotiated between peers. People have equal status.

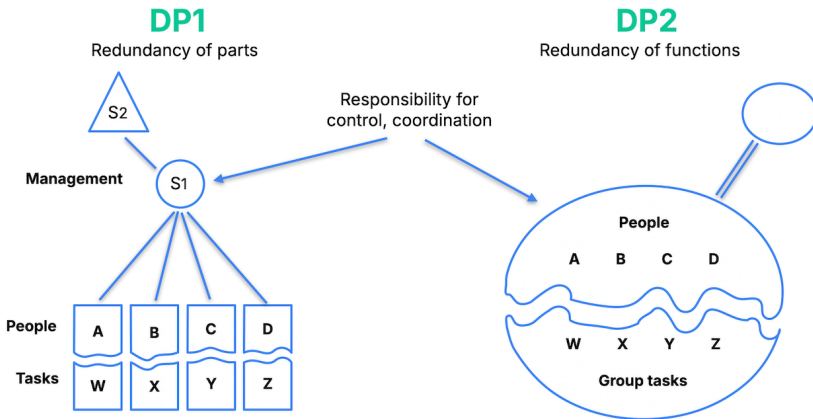


Figure 1. The two design principles. In DP1 (left), tasks W, X, Y, Z are each owned by individuals A, B, C, D under a supervisor S1 who handles coordination and control on behalf of management S2. In DP2 (right), the same tasks are owned collectively by the group, whose members coordinate themselves. Responsibility for coordination has moved from above the work to inside the group.

A few things worth pointing out before we move on. DP2 is not new and was not invented by Emery. It is how human beings have organised cooperative work for most of our history. The coal miners did not adopt DP2 in the 1940s; they returned to it. Hunter-gatherer bands, agricultural cooperatives, craft guilds, and the early modern shop floors before industrialisation. All of these were predominantly DP2 in structure. DP1 is not a recent invention either: the Roman Empire was governed by it, and royal courts have relied on dominant hierarchies for centuries. But it was only with the Industrial Revolution that ordinary people were forced into it through the factory system. Frederick Taylor then refined it into scientific management, and it now so dominates our experience of work that we mistake it for the natural order of things.

“It is only when the people involved work out their own designs that the necessary motivation, responsibility and commitment to effective implementation is present.”

– Merrelyn Emery³

³Merrelyn Emery, *Searching: For New Directions, in New Ways, for New Times* (Canberra: Centre for Continuing Education, Australian National University, 1982). The relevant section is available at socialsciencethatactuallyworks.com.

DP2 also assumes multiskilling is possible. Where it is not, the group adapts: control remains with the individual while responsibility for coordination and goals is shared across the group. This is relevant wherever deep specialist knowledge cannot be distributed, whether in transdisciplinary research, cross-functional product teams, or at C-level where each executive owns a distinct domain. It is the coordination and the goals that are shared, not necessarily the skills.⁴

DP2 groups can also organise internally as they see fit, including having a leader. A self-managing group that chooses one person to lead a particular piece of work, or to coordinate the group over time, has made that choice itself. The source of authority is the group, not the hierarchy above it. Leadership in DP2 is natural, but it is contextual and temporary: held at the group's discretion, not installed from outside. Most groups will want someone to coordinate, and that person may stay in the role for a long time. What makes it DP2 is not the absence of leadership but where the authority to lead comes from. A DP1 organisation cannot do the reverse: it cannot include genuine DP2 because the source of control is fixed above the work. DP2 contains DP1 as a choice. DP1 cannot contain DP2.

Finally, the two principles cannot be mixed. This claim sounds strong, and many readers will resist it. Surely there are gradations and combinations? Surely there is a middle ground? There is not. Responsibility for coordination and control is either with the people doing the work or it is not. A third state exists, but it is not a hybrid. It was first identified by Lewin, Lippitt and White and called *laissez-faire*: the absence of a design principle altogether.⁵ More recently, Don de Guerre and Merrelyn Emery identified modern forms of *laissez-faire* emerging in organisations that have tried to move toward DP2 without completing the transition.⁶ Both perform worse than even DP1. Let us take a closer look at this next, as many of the dysfunctions in this book emerge from exactly this state.

⁴Fred Emery and Merrelyn Emery, *Participative Design: Work and Community Life* (Canberra: Centre for Continuing Education, Australian National University, 1974), pp. 8–9. Available at socialsciencethatactuallyworks.com.

⁵Lippitt, R., & White, R. K. (1943). The “social climate” of children’s groups. In R. G. Barker, J. S. Kounin, & H. F. Wright (Eds.), *Child Behavior and Development* (pp. 485–508). Norwood, NJ: Ablex.

⁶Don de Guerre and Merrelyn Emery, “Modern Forms of Laissez-Faire Organization,” *Social Science That Actually Works*, 2008, socialsciencethatactuallyworks.com.

Modern forms of laissez-faire: the third state

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Why structure beats culture

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The six psychological requirements

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

In the Room

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The room that went quiet

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The phones came out

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The pair that runs everything

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The agenda that sabotaged itself

Context: The agenda looked good on paper. An opening keynote to align everyone on context, small group work to generate ideas, a panel of senior leaders to react and give feedback, and then a vote to prioritise. Energy is high at the start. The keynote is engaging. The groups work well for an hour. Then the panel arrives, and something shifts. Questions become pointed. A few people go quiet. The vote produces winners and losers rather than consensus. By the time people leave, the energy has curdled. The facilitator was experienced. The participants came in good faith. Nobody can quite explain what happened.

OST explains: What the agenda described was a textbook Mixed Mode event: a structure that alternates between DP1 and DP2 within the same session.

The keynote is DP1: an expert transmits to an audience. The group work is DP2: participants own the task. The panel reinstates DP1: authority figures evaluate what the groups produced. Merrelyn Emery was precise on this: mixed mode is not a synthesis or a balance between the design principles. Every time the design principle shifts within a session, it disrupts the group's sense of who is responsible, and Bion's basic assumptions are triggered fresh. The keynote primes dependency. The group work briefly generates genuine creative work. The panel reactivates hierarchy and invites fight or flight. This is not bad luck, and it is not a facilitation problem. It is the structural consequence of building incompatible design principles into the same event. The only reliable path is a workshop that stays on one design principle throughout. Prevention is straightforward. Recovery, once the room has turned, rarely is. The problem is not in the room. It is on the agenda.

* * *

The output nobody owned

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

HiPPOs and dungeon masters

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Tyranny of the majority

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The corridor conversation

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

In the Team

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The daily status report

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Passing the buck

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The powerless retrospective

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Forming–storming–norming–performing

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The hero culture

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The servant who still decides

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Working alone together

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Out of sight, out of sync

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The agile terrarium

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The collaboration that isn't

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The product owner trap

Context: The team has a product owner. She sits with the team, attends the standups, writes the user stories, and manages the backlog. She is good at her job, but she is not really making product decisions. She is translating decisions made by a business stakeholder who sits outside the team and controls the budget. The team builds what she tells them to build, being the proxy for the business which tells her to tell them. She does not really have the mandate to decide much. When the team raises a concern about direction, it goes to the PO, who raises it with the business, who considers it

and comes back three weeks later with a modified brief. The team is a feature factory. Everyone knows it. Nobody knows what to do about it.

OST explains: The product owner role, as conceived in Scrum, places a single person in charge of deciding what the team builds. That person is formally part of the team but functionally a conduit for management, the ordering-supplier divide between business and IT given a new job title. This is DP1: responsibility for goal-setting sits above the work, even if it sits only one step above rather than many. A genuinely self-managing group in DP2 owns its whole task, not just how the work is done, but what work is done and why. That means the team has a genuine relationship with the customer and the business context, not a mediated one through a proxy. Marty Cagan's distinction between feature teams and empowered product teams maps almost exactly onto DP1 and DP2: in the former, the team is handed a solution to build, in the latter, it is given a problem to solve.¹ The difference is not about process or methodology. It is about whether goal-setting lives inside or outside the group. Until that changes, the product owner is not empowering the team; she is the most visible evidence that it is not empowered.

* * *

Analysis paralysis

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The decision that went nowhere

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Permanent urgency

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

¹Marty Cagan, *Empowered: Ordinary People, Extraordinary Products* (Hoboken: Wiley, 2020).

Rearranging the furniture

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

In the Department

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

You are empowered now

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Psychological safety as a patch

Context: Your organisation may have realised that psychological safety is of the essence for good collaboration. Maybe it came out of a retrospective, maybe someone read the Google re:Work study and Amy Edmondson contributions,¹ or maybe it was a leadership initiative after too many people stopped speaking up in meetings. Either way, there are now workshops on it, a section in the onboarding, maybe even a survey to measure it. Leaders are coached to create it. Teams are encouraged to demand it. And yet, somehow, people are still not speaking up, still not taking risks, still not challenging the decisions made above them. The patch does not seem to be holding.

OST explains: Psychological safety is real, and it matters a lot, but what is often missed is that it is an emergent property of the structure people work in, not something you can install. In a DP1 organisation, people are inherently in a dependent, subordinate position, and the rational response to that is to be careful about what you say and to whom. That is not a personal failing; it is Bion's basic assumptions playing out exactly as expected: dependency, fight/flight, and factionalism are the natural human response to autocratic hierarchies. You cannot train people out of that while the structure that causes it remains intact. In a DP2 structure, on the other hand, psychological safety is not a programme or a value on the wall; it is simply what happens

¹Amy C. Edmondson, *The Fearless Organization: Creating Psychological Safety in the Workplace for Learning, Innovation, and Growth* (Hoboken, NJ: Wiley, 2018). The Google re:Work findings are summarised at rework.withgoogle.com.

when people are peers designing and owning their own work. They speak up because it is their job to, and because there is no hierarchy of dominance to be careful around. Demanding psychological safety in a DP1 organisation is a bit like applying a patch to a system with a structural bug. It might cover the symptom for a while, but the underlying code has not changed.

* * *

DORA, the wrong way round

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Team Topologies, the wrong way round

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Involvement theatre

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Better leaders is the answer

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Change agents of the status quo

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The external verdict

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The frozen middle

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

It's a communication problem

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Them and us

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Fixing people

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The error factory

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Burned by design

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Designed to undermine

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The pilot trap

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Fixing the process

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Local optimisations

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

In the Organisation

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

OKRs imposed from above

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Budgets are bureaucracy

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The performance review

Context: It is that time of year. Everyone knows it is coming. The forms go out, the self-assessments are written, the manager prepares their ratings, and the one-to-ones are scheduled. The organisation may have moved to agile, reorganised around teams, and invested in collaboration, but for these few weeks, the focus shifts unmistakably back to the individual. People remember that their pay, their career, and their standing in the organisation are determined not by how the team performed but by how they personally look to the person above them. The careful positioning begins. The collaborative spirit that seemed genuine a month ago quietly recedes.

OST explains: The annual individual performance review is one of the most structurally destructive rituals in modern organisations precisely because it is so normalised. Whatever DP2 conditions the organisation has managed to create during the year, the performance review reinstates DP1 values in a single stroke: you are an individual, your performance is separable from the group's, and someone above you is qualified to judge it. In a genuinely self-managing group, individual performance reviews are not only unnecessary,

they are actively harmful because they reintroduce personal dominance through one of its most intimate mechanisms. Research consistently shows that external individual evaluation undermines exactly the intrinsic motivation that DP2 structures produce. The review is not a neutral administrative process. It is a structural signal about who is really in charge and what actually gets rewarded.

* * *

The learning organisation that doesn't learn

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The short-termism machine

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The strategy that did not survive contact with reality

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The leadership team that isn't

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Quiet quitting

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The Sunday email

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

It's just a job

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The project in product clothing

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Pay and reward

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Career paths

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The agile scaling trap

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Intermezzo: Objections

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Self-management doesn't scale

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Nobody actually does this

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

All teams need a manager

The objection comes in several forms. Without a manager, the team will descend into groupthink. Without a manager, nobody coordinates. Without a manager, conflicts fester. Different arguments, same premise: teams cannot function without someone in authority above them.

Take groupthink. The claim is that without a manager, teams suppress dissent and converge on poor decisions. But groupthink is a product of anxiety and hierarchy, not self-management. It occurs when the cost of disagreeing is too high. That is a description of DP1. In a genuinely self-managing group, where people have designed their own work and share accountability for outcomes, individual opinions are the point. The structure requires them. The structure that reliably produces groupthink is the one with a single point of authority whose approval everyone needs.

Coordination is not absent in DP2. It is differently located. The group owns it, which means the group can choose from within it a coordinator who tracks dependencies, calls the group together, and keeps the process moving. That person serves the group. They do not supervise it. A coordinator the group selects and can replace is a function. A manager the organisation installs above is a design principle. The distinction is not semantic.

Conflicts the group can handle. When it cannot, it can call on a facilitator or mediator. None of this requires a permanent authority above the group.

What the objection really fears is not groupthink or coordination failure or unresolved conflict. It is the absence of a visible person to blame when things go wrong. That is a DP1 need, not a structural one.

* * *

This is fifty-year-old theory

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

This sounds like communism

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

IT is different

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Agile works fine, why do we need OST?

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Developers just want to code

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

We've already transformed

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

What the Open System Does

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Where the world is now

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Active and passive adaptation

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Beyond the Walls

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Built for yesterday

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Deploying AI into a broken system

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The customer we never met

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The AI we cannot talk about

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The market we think we shape

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Outsourcing the future

Context: Your organisation wants a new strategy. Leadership does not have the bandwidth, or perhaps the confidence, to lead the process internally. A consulting firm is brought in. They are smart, experienced, and methodologically rigorous. They interview the leadership team, benchmark against competitors, analyse market trends, and run workshops with selected stakeholders. A few months later a strategy document arrives. It is well-structured and credible. The organisation begins implementing it. Some things work. Others do not land the way the consultants predicted. The people doing the implementation have questions the document does not answer. The consultants are gone. Two years later a new consulting firm is brought in to develop the next strategy.

OST explains: Strategy is not a document. It is a shared understanding of where a system is, where it needs to go, and why. That understanding cannot be imported from outside because it depends on knowledge that only exists inside the system, distributed across the people who operate it, the relationships they hold, and the environment they transact with daily. A consulting firm can bring analytical frameworks, comparative data, and an external perspective. What they cannot do is replace the participative process through which an organisation develops genuine strategic ownership. The Search Conference exists because Emery understood this. An effective strategy requires the whole system in the room, not a representative sample interviewed by outsiders. A strategy people helped build is one they will adapt when reality diverges from the plan. A strategy handed down, however well-researched, will be executed literally until it fails and then replaced by the next one. The cycle is not a strategy problem. It is a participation problem.

* * *

People resist change

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Doing the wrong thing right

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The Way Out

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The Two-Stage Model for Structural Change

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The Search Conference

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

The Participative Design Workshop

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Epilogue: The Diagnosis

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Glossary

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

Acknowledgements

This content is not available in the sample book. The book can be purchased on Leanpub at <https://leanpub.com/organisationaldysfunctions>.

About the Author

Trond Hjorteland is an IT architect and sociotechnical systems practitioner at Capra Consulting, where he helps organisations design themselves for a turbulent world. Drawing on open systems theory and its research tradition, he bridges domain-driven design, event-driven architecture, and human-centred organisation design.

He holds a Cand. Scient. in astrophysics from the University of Oslo. His professional background is in large, complex, business-critical IT systems, as a developer, architect, and advisor across telecom, media, and the public sector. Over the past decade, he has become equally focused on the organisational dimension of that work: how structure shapes behaviour, why agile so often disappoints, and what the research actually shows about what works. That has led him to Open Systems Theory and sociotechnical systems design, including working directly with Merrelyn Emery, one of the field's original pioneers, on a large-scale survey of the IT industry.

He is a board member of STS Kompetansenettverk and a regular speaker at conferences including KanDDDinsky, DDD Europe, NDC Oslo, and Fast Flow Conf. Recordings of many of his talks are available on [YouTube](#).

He writes at [linkedin.com/in/trondhjort](https://www.linkedin.com/in/trondhjort) and on Mastodon at hachyderm.io/@trondhjort. His personal site is trond.hjorteland.com (a 90s relic).