

Organization

What Institutions Can Learn from Software Engineering and What They Cannot

Kai Froeb



About this translation

This translation was produced by machine using AI and has not yet been fully checked by a human. If you find any errors, please report them to us at feedback@reflexivitypress.com. Anyone who has bought the book and reports errors to us will receive the corrected version as an e-book free of charge.

Author

Kai Froeb

https://kai.froeb.net

Publisher

Reflexivity Press

São Martinho do Porto

Kai Froeb Edition

Latest version and notes on corrections:
https://reflexivitypress.com/en/books/froeb-organisation

Feedback: feedback+en.organisation.v1.5@reflexivitypress.com

Cover: https://reflexivitypress.com/en/about/cover-art/

Publication

2026-11-21

Translation: 2026-11-22

Version 1.5

Typeset with LaTeX (LuaHBTeX, Pandoc + Eisvogel)

Note on authorship

Written with the language model Claude (Anthropic). Conception, content and responsibility rest with the author. The translation was made with the help of AI language models. How the work was done is described in the essay “Philosophy in the Age of Machines That Write”: https://reflexivitypress.com/en/about/machines-that-write/

© Copyright 2026 by Reflexivity Press, São Martinho do Porto.

All rights reserved.

Zero Marginal Cost License 1.4 (ZMC 1.4)

Reader-friendly summary

What has no marginal cost should have no barriers.

You may

- **Read, download, and copy** the work free of charge.
- **Share the unaltered work digitally for noncommercial purposes**, with attribution to the author, source, and license.
- Convert the file format and make technically necessary accessibility adjustments without changing the content.
- **Print or commission individual copies and give them away** for personal, noncommercial use; direct costs may be reimbursed without markup or profit.
- **Translate the work** for your own personal, nonpublic, noncommercial use.
- **Mirror the unaltered work** openly: noncommercially, with version, date, and a reference to the current version.
- Use the work individually for education and research, and quote it for noncommercial purposes in accordance with scholarly standards, without using a collection of excerpts as a substitute for the work.

Conditions

Use must remain **noncommercial**. When sharing the work publicly, identify the **author, source, and license**, wherever reasonably possible. **Only the rights holder may relicense the work or issue new versions.** You must not impose additional legal or technical restrictions or suggest endorsement by the author or publisher.

All other rights reserved

This license permits only the uses expressly stated. Any use beyond these permissions requires separate authorization from the rights holder, unless it is already permitted by law.

Statutory permissions and remuneration claims remain unaffected. Third-party material may be subject to separate terms.

Important: This summary is for guidance only and is not a standalone license. The full **ZMC 1.4** is the authoritative text:

https://reflexivitypress.com/en/zmc/



About the Author

Kai Froeb (born 1963) is a German philosopher and IT expert living in São Martinho do Porto, Portugal. His work combines more than 35 years of systematic Hegel research with the possibilities of artificial intelligence, developing a new form of philosophical collaboration.

His thinking is radically interdisciplinary. He studied mathematics and computer science at the University of Erlangen–Nuremberg, business administration at EBS, and cultural studies and law at the University of Hagen. At the same time, he pursued intensive general studies in philosophy, sociology and history. He deliberately conducted his studies without orienting them towards academic degrees, preserving the freedom to follow an independent path of thought and research beyond the boundaries of university disciplines.

Since the late 1990s, Froeb has edited and maintained **hegel.net**. Available in twelve languages, the website ranked among the leading results in international Google searches for Hegel for many years and developed into one of the central online resources on Hegel's philosophy. It is probably still the most extensive multilingual website devoted to Hegel on the internet.

In connection with **hegel.net** and **hegel-system.de**, Froeb also created an extensive, publicly accessible archive of primary sources. The German-language collection alone offers 506 digitised original editions of works by Hegel and nineteenth-century Hegelians for free download. Further collections in other languages include 19 original works in English.

More than 40 years of programming experience, beginning in 1980, form another methodological foundation of his work. His first program was a reimplementation of Joseph Weizenbaum's ELIZA, thus completing a circle from the earliest experiments in artificial intelligence to his productive collaboration with Claude today. Froeb uses his IT expertise not merely as a technical resource but also as an instrument of thought: he analyses social and philosophical structures with the precision of a software architect.

Froeb is developing an independent philosophical body of work that stands in the tradition of Hegel while attempting to continue that tradition under contemporary conditions. At its centre is his own performative approach to truth, which forms the common foundation of his philosophical books. Truth appears here not merely as the object of propositions, but as a movement that must also prove itself in the form of thinking and presentation.

His aim is demystification through conceptual clarification: to penetrate every boundary through thought, including the boundary of what initially appears inaccessible. In his work, form follows content—structure, concepts and method are determined by the subject matter itself. At the same time, he seeks accessibility: to present demanding philosophy as clearly as possible, in the spirit of the best Anglo-American scholarly tradition, without simplifying it.

His guiding motif is Hegel's formula: "The truth is the whole"—the search for a form of integration that need not repress or exclude anything.

CONTENTS

Preface 1

Chapter 1: What an Institution Is For 3
- 1.1 The End . . . 3
- 1.2 Dead Code . . . 3
- 1.3 Where the Transfer Breaks . . . 4
- Summary of Chapter 1 . . . 5

Chapter 2: Dividing 7
- 2.1 Responsibility . . . 7
- 2.2 Interfaces . . . 7
- 2.3 Where the Transfer Breaks . . . 8
- Summary of Chapter 2 . . . 9

Chapter 3: One Thing, One Place 11
- 3.1 The Copy That Drifts Apart . . . 11
- 3.2 What the Rule Achieves . . . 11
- 3.3 Where the Transfer Breaks . . . 12
- Summary of Chapter 3 . . . 12

Chapter 4: As Simple as Possible 13
- 4.1 Procedures One Understands . . . 13
- 4.2 The Fence in the Field . . . 13
- 4.3 Where the Transfer Breaks . . . 14
- Summary of Chapter 4 . . . 14

Chapter 5: Knowledge and Decision 15
- 5.1 The Briefing Paper . . . 15
- 5.2 Weighting and Overriding . . . 15
- 5.3 What Each Owes the Other . . . 16
- 5.4 Where the Transfer Breaks . . . 17
- Summary of Chapter 5 . . . 17

Chapter 6: Measuring and Testing 19
- 6.1 Feedback . . . 19
- 6.2 When the Measure Becomes the Target . . . 19
- 6.3 Measuring Without Steering . . . 20
- Summary of Chapter 6 . . . 21

Chapter 7: Errors **23**
7.1 Showing Errors . . . 23
7.2 Looking Into It Instead of Looking for Culprits . . . 23
7.3 Where the Transfer Breaks . . . 24
Summary of Chapter 7 . . . 24

Chapter 8: Versions and Transitions **25**
8.1 Do Not Overwrite . . . 25
8.2 The Memory of an Institution . . . 25
8.3 Transitions . . . 26
8.4 Where the Transfer Breaks . . . 26
Summary of Chapter 8 . . . 27

Chapter 9: Rules That a Machine Executes **29**
9.1 Like Cases Alike . . . 29
9.2 The Case the Rule Did Not Foresee . . . 29
9.3 Where the Transfer Breaks . . . 30
Summary of Chapter 9 . . . 30

Chapter 10: Openness **31**
10.1 Reasons One Can Read . . . 31
10.2 Openness Binds . . . 31
10.3 Where the Transfer Breaks . . . 32
10.4 Who Controls Others . . . 32
Summary of Chapter 10 . . . 33

Chapter 11: Working with Language Models **35**
11.1 What Has Become Cheap . . . 35
11.2 What a Model Can and Cannot Do . . . 35
11.3 Two Dangers That Reach Further . . . 36
Summary of Chapter 11 . . . 37

Chapter 12: How Much Effort **39**
12.1 Proportionate . . . 39
12.2 Examining One’s Own Rules . . . 39
12.3 The Machine Made of People . . . 40
12.4 When the Rules Contradict One Another . . . 41
Summary of Chapter 12 . . . 41

Chapter 13: Why Reason Has a Chance **43**
13.1 Making the Right Thing Easy . . . 43
13.2 Hegel’s Hope . . . 44
13.3 Whose Advantage . . . 44
Summary of Chapter 13 . . . 45

Conclusion: Where One Can Begin **47**
One Process Instead of a Reform 47
Seven Questions 47
What One Can Do Where One Stands 48

PREFACE

Anyone who works in a public authority, a company, a hospital, or an association knows the complaints: no one feels responsible, the same information is requested three times, reports are written for no one, decisions are made without those who know. Most people take this to be unavoidable or to be the fault of individuals. This book takes it to be questions of design that can be answered, and it looks for the answers first in an unexpected place.

Over half a century, software engineering has learned to build large systems with a division of labor so that they can be understood, checked, and changed. It had to learn this because programs do not show their errors of their own accord. A program can calculate wrongly in rare cases for years without anyone noticing, and whoever built it talks himself into thinking it fine as readily as anyone else. That is why software engineering has developed procedures that make errors visible before they do harm, and that prevent one person alone from judging the quality of his own work. What it has found are not rules for computers but rules for the division of labor. Many of them can be transferred to institutions in which people work together.

Every transfer, however, breaks at a particular point. The parts of a program want nothing; the people in an institution have purposes of their own, are at the same time parents, neighbors, and citizens, and reflect on their work. That is why each chapter asks two things: what a rule of software engineering achieves in organizations, and where it breaks on people. At this second point, it is no longer technology that decides, but what the institution owes to those who work in it and to those whom it affects.

The book presupposes no specialist knowledge, either of software or of organizations. It is an extension of *Ethics – The Good We Already Do* (2026), the book about what we owe one another, and asks what follows from it for institutions in which people work together. Alongside it stand *Truth* (2025) on checking and *Thinking – Ordering Without Losing* (2025) on ordering, which proceed in the same way. What it needs from these books, it develops itself. At the end of each chapter there is a summary, and anyone short of time can read the book through the summaries.

CHAPTER 1: WHAT AN INSTITUTION IS FOR

1.1 The End

A citizens' office exists so that people obtain the identity cards, registration certificates and permits to which they are entitled. That includes checking whether they are entitled to them, protecting identities and treating like cases alike; an office can do good work by turning an application down. A hospital exists so that the sick are treated safely and well, with means that are not unlimited; a sports club, so that people can play sport. Obvious as this sounds, it is easily lost from view. The clerk in the citizens' office who asks for a third document is thinking of the regulation, not of the woman who has to take another day off work to bring it.

Everything an institution does is a means to its end: its departments, its forms, its meetings, its rules. A means is as good as it serves the end. From this follows the question that every rule and every step of work must be able to answer: What is this for? Whoever cannot answer it has either forgotten something, or the step is superfluous.

1.2 Dead Code

In software engineering, a part of a program that nothing calls any longer is called dead code. It does nothing harmful, but it has to be read, maintained and taken into account with every change, and whoever comes across it often does not know whether he may delete it. Good programmers remove it as soon as they are sure that nobody needs it.

Institutions have the same thing: the statistic that a minister once wanted who has long since left office, the committee whose occasion has been forgotten, the department that has lost its task and carries on. Some of it even grows. A unit that has lost its task looks for new ones so as not to be dissolved, and

whoever wants to hand off work would rather acquire subordinates than equals. So an agency grows while its task shrinks. The British historian Cyril Northcote Parkinson described this in the 1950s, among other cases in the British Colonial Office, whose staff grew steadily from 1935 onward and was at its largest in 1954, the last year he counted, while the colonial empire was shrinking.[1]

1.3 Where the Transfer Breaks

A program has its end from the one who writes it. An institution has it from many: from those who founded it, from those who pay for it, from those who work in it, and from those for whom it exists. These ends do not agree of their own accord. For the employees, the institution is also the place where they earn their living; for the management, also the place of their standing; for the funder, also an item in the budget. That is why the end of an institution can be taken over by other ends without anyone deciding it. In the end the citizens' office serves above all its own procedures and the hospital above all its balance sheet.

This cannot be prevented by driving their own ends out of people; they are the ones who do the work. It can only be limited by the institution making its end explicit and measuring everything it does against that end, again and again. A program cannot do this; it executes what it has been told to do. An institution that stops measuring its rules against their end becomes a machine made of people. That is what is called bureaucracy in the bad sense. In the good sense, bureaucracy means that decisions are made according to rules, responsibilities and files instead of mood and connections, and compared with arbitrariness that was a great step forward. The evil is not the rule but its becoming independent of the end. This examination of one's own rules runs through the whole book; Chapter 12 asks how much of it one must afford.

Looked at closely, three things are to be distinguished here: the declared end of an institution, the ends of the people who work in it and deal with it, and what it actually brings about. In a company they lie far apart. It is supposed to serve customers, earn revenue, bring the owners a profit and secure the employees their living, and these ends do not agree of their own accord. To this comes a second question, which comes before the first. An institution can fulfill its end excellently and yet serve an end that does not hold up before those it affects. Whoever judges an institution therefore asks two things: Does it fulfill its end?

1 C. Northcote Parkinson, "Parkinson's Law" (DE: „Parkinson's Law"), *The Economist*, November 19, 1955; expanded in *Parkinson's Law and Other Studies in Administration* (1957).

And is this end, together with the means by which it pursues it, justified before all whom it affects? The two cannot be answered quite separately, for how an institution pursues its end is part of what must hold up before those affected. This book deals above all with the first question. It does not forget the second, for a well-built institution that does harm only does it more thoroughly.

The opposite answer is dangerous too: having every unit constantly prove its usefulness, in metrics, quarter after quarter. Then the institution is occupied with the proof instead of the end, and whatever cannot be counted disappears (Ch. 6).

Further reading: *Ethics – The Good We Already Do* (2026), Ch. 12 (institutions preserve reasons), *Capitalism — An Introduction* (in preparation), Ch. 29.6 (the momentum of bureaucracy), *Capitalism — An Introduction*, Ch. 29.10 (the state as a company)

Summary of Chapter 1

An institution exists for an end, and everything it does is a means to it. Every rule and every step must therefore be able to say what it is for. Whatever cannot is like dead code in a program: it costs maintenance and blocks the view. Unlike a program, an institution has its end from many, and that end can be taken over by their own ends without anyone deciding it. What helps against this is not driving people's ends out of them but making the institution's end explicit and measuring its rules against it again and again. An institution that gives this up becomes a bureaucracy in the bad sense, in which the rules have become independent. But the question is not only whether it fulfills its end, but also whether this end holds up before all whom it affects.

CHAPTER 2: DIVIDING

2.1 Responsibility

Whoever cannot manage a large task alone divides it. The question is how. A club that holds a summer party can distribute the work by persons (“Anna does what Anna can do” (DE: „Anna macht, was Anna kann“)) or by tasks (food, music, permit, clearing up). The first division works as long as Anna is there. The second also works next year, because everyone knows where to ask.

Software engineering calls this separation of concerns. Each part of a program is to have a clearly defined task, so that a change stays in one place as far as possible. Whoever wants to know how the invoice is produced finds it in one place; whoever wants to change it changes it there, without touching the rest. This never quite succeeds, since the parts have to work together, but the closer one comes to it, the easier the whole is to understand and to change. For institutions this means: for every task there is a unit that carries it, and everyone can know which one it is. Where two feel responsible, they quarrel; where nobody feels responsible, the matter is left lying.

2.2 Interfaces

Between the units things are handed over: a file, an order, a figure. Software engineering attaches great importance to these handovers being described exactly. What one part receives from another and what it returns is fixed, and how it does this internally is none of the others’ business. Then one part can be improved without all the others having to be changed along with it.

In institutions the handovers are often the weakest points. The ward hands the patient over to the rehabilitation clinic, and half of what the ward knew does not arrive. The building authority passes the application on to the environmental agency, and neither knows what the other has already checked. A good handover says what is being handed over, what the recipient is to do with it and what is still

open. That costs a few minutes at each handover and saves the hours in which someone later asks back, or does not ask back.

How strongly handovers shape an institution is shown by an observation that Melvin Conway made about software development in 1968: organizations build systems that mirror their own communication structure.[1] If three departments work on a program, it has three parts, whether the matter requires it or not. The same holds for laws that come from three ministries and for forms that serve three agencies.

2.3 Where the Transfer Breaks

In a program, responsibilities can be cut as the matter requires and cut anew when the matter changes. In an institution, people, posts and budgets hang on every cut. Responsibilities are cut once and remain, even when the tasks shift. Above all, however, the connections within the matter do not keep to them.

A new road affects traffic, the air, health, noise and the budget. Each agency sees in it what it is responsible for, and each examines it thoroughly. What lies between the responsibilities none of them examines: that the road attracts new traffic, which makes the air worse, which the public health office notices only in its statistics, years later. Whoever brings the parts together only at the top, in the meeting of agency heads or in the cabinet, sees only what one of the agencies has already seen. The order of responsibilities is a tree in which each task has exactly one place; the connections within the matter are a network.

Good institutions therefore lay a network over the tree: joint working groups for questions that concern several units, the duty to involve others, and units that look across all the others at the whole. How well this network holds is not decided only in the individual case but by which connections have been provided for.

The network too can become an end in itself. Every new round of coordination and every further duty to involve others has its reason, and yet with their number an institution does not become more coherent but more fragmented. Each unit sits in so many rounds that it no longer thinks in any of them, and whatever no one wants to decide alone is passed upward until it arrives there as the lowest common denominator. Everything is then connected, and nothing is thought through together. A good network has few connections, but at the points where

1 Melvin E. Conway, “How Do Committees Invent?” (DE: „How Do Committees Invent?“), *Datamation* 14, no. 4 (April 1968), pp. 28–31.

the matter requires them. And there remains the question of who decides what. The simple answer is: as close to those affected as possible, as high up as necessary. The more precise one adds that a question belongs where everyone it affects is heard, not only those who sit at the table.

Further reading: *Thinking – Ordering Without Losing* (2025), Ch. 9 (dividing by material and by function), *Thinking – Ordering Without Losing*, Ch. 10.3 (tree and network), *Ethics – The Good We Already Do* (2026), Ch. 12.8 (who regulates what), *Hegel's Philosophy of Right – How Freedom Becomes Real* (2026), Ch. 29.4 (branches that are meant to converge at the top)

Summary of Chapter 2

A task that one person cannot manage alone is divided, and better by tasks than by persons. For every task there is a unit that carries it, and everyone can know which. Between the units things are handed over, and a good handover says what is being handed over, what is to be done with it and what remains open. Organizations build their results in the image of their own division. The connections within the matter do not keep to the responsibilities: what lies between them no unit examines, and at the top one sees only what one of them has already seen. That is why the tree of responsibilities needs a network of connections, but only where the matter requires them; a network made of nothing but rounds of coordination fragments the institution instead of connecting it. A question belongs where everyone it affects is heard.

CHAPTER 3: ONE THING, ONE PLACE

3.1 The Copy That Drifts Apart

A family moves house. It reports the new address to the citizens' office, the health insurer, the tax office, the school, the bank, the insurance company. Half a year later a letter arrives at the old address because one office was forgotten, and a year later the address at the school is wrong because of a typing error there. Each office has a copy of the same piece of information, and copies drift apart.

Software engineering has one of its best-known rules for this: every piece of information should have exactly one place where it is valid; all others fetch it from there. If it is changed, then in this one place, and everyone sees the new version at once. The same rule holds for knowledge that is written down in texts. A procedure described in three manuals is changed in one of them at the next revision and not in the other two, and from then on nobody knows which one is valid.

3.2 What the Rule Achieves

In institutions the rule not only saves work, it prevents contradictions. Two regulations that answer the same question differently almost always arise because the same matter was regulated in two places and only one of them was brought up to date. Whoever looks for a rule should find it in one place; whoever changes it should know that he thereby changes it for all cases.

A small publishing house that produces books in many languages with its own tools applies the rule strictly. The translation of a technical term stands in exactly one list, and all translations fetch it from there. Files that a program generates from other files are never changed by hand, for the change would be lost at the next run and the source would no longer be correct. The rule takes effort at the beginning, because for every piece of information one has to decide where it belongs. After that it spares one the search for the contradiction.

3.3 Where the Transfer Breaks

In an institution, duplicate information is sometimes not an oversight but a safeguard. Accounting keeps its figures, and the internal audit checks them with its own records, precisely because it does not use the same ones. For important payments, two people sign. A court does not rely on the police file but hears the witnesses itself. Where control matters, a second, independent version of the same information is intended; it is the point of the control.

The central file has a reverse side too. A single file for every citizen, from which all agencies draw, would be clean as data management. It would at the same time be an instrument of power that nobody should have who could misuse it, and that means: nobody. That the agencies do not know everything about one another is a nuisance for someone who moves house and a protection for someone who is persecuted.

Moreover, not every second piece of information about the same thing is a copy. The old address is correct for the past, the new one for today; the address someone gives can differ from the one the agency has established. These are different statements with their own claim to validity, not copies that have drifted apart.

For institutions the rule therefore reads more precisely: for every question there is one source that is authoritative, and copies exist only where they have an end of their own, control or protection. Then every duplication is justified, and all others fall away.

Further reading: *Thinking – Ordering Without Losing* (2025), Ch. 4 (orders that know nothing of the matter), *Hegel's Philosophy of Right – How Freedom Becomes Real* (2026), Ch. 29.8 (how a power binds itself)

Summary of Chapter 3

Copies of the same information drift apart. That is why there should be one authoritative source for every question, and all others should fetch the answer from there. This saves work and prevents contradictions, which almost always arise because the same matter was regulated in two places. In institutions, however, a second version is sometimes intended: as independent control, and the separation of files as protection against a power that knows everything. Duplications therefore exist only where they have an end of their own.

CHAPTER 4: AS SIMPLE AS POSSIBLE

4.1 Procedures One Understands

A form with forty fields is filled in wrongly, one with ten correctly. A regulation that everyone understands is followed; one that first has to be looked up is circumvented or, out of fear, over-fulfilled. Simplicity is therefore not a convenience but a condition for rules to have any effect.

Software engineering says the same of programs: what is built simply has fewer errors, and the ones it has can be found. To this comes an observation that John Gall put pointedly in 1975: a complex system that works has invariably grown out of a simple system that worked; one designed to be complex from the start never works and cannot be patched up to make it work.[1] For institutions this means: start small, see what holds, and only then expand.

4.2 The Fence in the Field

Whoever wants to simplify comes upon rules whose point he does not see. The writer Gilbert Keith Chesterton found an image for this: a reformer finds a fence across a road and wants to tear it down because he sees no use in it. A wiser one tells him he should first find out why the fence was built; then he may tear it down.[2]

Many rules preserve a reason that no one knows any longer: an accident, a fraud, a dispute that was not to happen again. Whoever deletes them without knowing the reason brings the accident back. That is why the first question in simplifying is the same as in Chapter 1: What is this for? Only when the answer is that the reason has fallen away or can be achieved more simply may the rule go. If the reason cannot be found out, that is itself a finding: an institution that does

---

1 John Gall, *General Systemantics* (1975), later *Systemantics*. Paraphrased.

2 Gilbert Keith Chesterton, *The Thing* (1929), chapter “The Drift from Domesticity” (DE: „The Drift from Domesticity“). Paraphrased.

not record the reasons for its rules will either fear every simplification or regret it (Ch. 8).

4.3 Where the Transfer Breaks

In a program the simpler solution is usually also the better one. In institutions simplicity can shift costs. An application that is simple for the agency can be hard for the applicant, and vice versa. The flat rate simplifies administration and disadvantages the person whose situation it does not fit. The question is not whether something is simple but for whom.

Nor is simplicity always compatible with justice in the individual case. A rule that considers every case is complicated; a simple one gets some cases wrong. Good institutions resolve this by having a simple rule for the ordinary case and an orderly path for the exception: who can apply for it, who decides, and that the decision is given reasons (Ch. 9).

Further reading: *Ethics – The Good We Already Do* (2026), Ch. 12.3 (form and content), *Ethics – The Good We Already Do*, Ch. 12.4 (red at night), *Thinking – Ordering Without Losing* (2025), Ch. 15 (dead order)

Summary of Chapter 4

Procedures that people understand are followed; that is why simplicity is a condition for rules to have any effect. What works has almost always grown out of something simple, and that is how institutions should grow too. Whoever simplifies must first know what a rule is for, since many preserve a reason that no one knows any longer. But simplicity can shift costs, from the agency to the citizen or to the person whose case the flat rate does not fit. That is why one must ask for whom something is simple, and besides the simple rule an orderly path for the exception is needed.

CHAPTER 5: KNOWLEDGE AND DECISION

5.1 The Briefing Paper

A hospital has to decide whether to keep its maternity ward, where fewer and fewer children are born. The managing director cannot gather the figures, the situation of the midwives and the travel times to the next hospital herself. She has the ward and the administration each write her a briefing paper. What is in them is almost everything she knows when she decides.

So it is wherever more people work together than one person can survey: whoever decides knows less than those who prepare the decision. That is the point of the division from Chapter 2. The briefing paper is the handover at which knowledge changes hands, and what is lost there no decision brings back.

Software engineering demands three things of every such handover. A part says when its result is uncertain, instead of silently delivering a plausible figure with which everything further on calculates unawares. Where much is at stake, a second party checks who does not want the same thing and does not have the same error in mind. And what happened is recorded with its reason, so that later one can find where an error came from. All three rules apply to briefing papers too. Only, the parts of a program have no ends of their own, and the people who write briefing papers do.

5.2 Weighting and Overriding

The ward wants to stay, often for good reasons. The administration is measured by costs. Neither writes anything false: the ward gives the travel time on black ice, the administration the one on an ordinary morning; the ward counts the births of the last two years, the administration the average over ten. Every figure is correct; what has been selected is whatever supports one's own conclusion.

This usually happens out of conviction, not calculation, and is therefore hard to recognize.

The managing director knows this and discounts the presumed interest. She has reason to, for she sees what no department sees: where the money will be missing elsewhere, what the hospital's owner demands. But together with the presumed interest she strikes out the knowledge contained in the briefing paper. Perhaps the travel time on black ice was precisely the case that matters. And she has ends of her own: to look good, to avoid a conflict. From outside it is hardly possible to tell whether she overrides out of mistrust or because the conclusion does not suit her.

Both sides learn. The ward writes its warnings more strongly because they are halved; the management halves more strongly because they are exaggerated. In the end each side calculates with what the other will make of its information, and the institution pays for its knowledge without using it. Neither side can stop this alone. Whoever alone weights honestly is heard too little, and whoever alone stops halving is deceived.

5.3 What Each Owes the Other

Whoever prepares a decision owes the one who makes it the whole picture, as far as it can be relevant to the decision. That does not mean knowing everything, but suppressing nothing essential merely because it speaks against one's own proposal, and naming the uncertainties one knows of. Whoever decides owes it to the matter to hear this, and owes those whom the decision affects a reason where he departs from it. The two duties depend on each other: a complete briefing paper is worth writing only if it is read, and a justification only if the briefing paper was complete. That is why a form must bind both at once.

Such a briefing paper names the options that were examined and rejected, the counter-reasons, and how certain the figures are; then the management can ask questions instead of halving. A unit does not report alone on a matter to which it is itself attached. A second view comes from a unit that gains nothing from the outcome, and where that is not possible, at least it is stated openly what those who prepared the paper stand to gain from the decision. Whoever was of a different opinion does not disappear from the paper, for it is precisely his doubt that is the knowledge the management needs. Finally the decision is recorded together with its assumptions, and after a year one checks whose information

was correct. That rewards the side that was right, not the one that weighted most loudly.

5.4 Where the Transfer Breaks

The interest that weights the briefing paper cannot be abolished, and one should not want to abolish it. The ward is attached to its ward, and precisely for that reason it knows things about it that no one else knows; indifferent briefing papers are not more accurate but poorer. An institution can only make the interests visible and bring them into balance against one another.

This too presupposes that the management wants to see the whole. Where it only secures its own position, the forms become a backdrop: dissenting votes are recorded and not read, departures are justified with sentences that need not hold up before anyone. The only thing that helps against this is that the reasons become public, before those whom the decision affects (Ch. 10).

Max Weber described one side of this relationship: administration exercises "domination through knowledge" (DE: „Herrschaft kraft Wissen") and likes to keep this knowledge secret.[1] Friedrich August von Hayek emphasized the other: the knowledge that matters is dispersed among many, and no management ever has all of it.[2] A good institution neither holds knowledge fast at the top nor leaves it lying at the bottom. It builds the path along which knowledge travels so that one can see along the way who weighted it and how.

Further reading: *Ethics – The Good We Already Do* (2026), Ch. 15.8 (what preparing and deciding owe each other), *Truth* (2025), Ch. 19 (how to weight without lying), *Hegel's Philosophy of Right – How Freedom Becomes Real* (2026), Ch. 29.4 (expertise and leadership in government), *Capitalism — An Introduction* (in preparation), Ch. 29.6 (specialist department and management in the corporation)

Summary of Chapter 5

Whoever decides knows less than those who prepare the decision, and in the briefing paper knowledge changes hands. Those who prepare it weight it, often out of

1 Max Weber, *Wirtschaft und Gesellschaft*, Tübingen 1922, Part I, ch. III, § 5, p. 129.

2 Friedrich August von Hayek, "The Use of Knowledge in Society" (DE: „The Use of Knowledge in Society"), *American Economic Review* 35, no. 4 (1945), pp. 519–530. Paraphrased.

conviction; the management discounts the presumed interest and strikes out the knowledge along with it. Because both learn, in the end each calculates with the other's reaction, and the institution no longer uses its knowledge. Whoever prepares owes everything relevant, including what speaks against his own proposal; whoever decides owes a hearing and a reason where he departs from it. Forms bind both: rejected options and counter-reasons in the briefing paper, a second view without an interest of its own, recorded dissenting votes, the check after a year. The interest that weights the briefing paper also makes it knowledgeable; it cannot be abolished, only made visible and balanced.

CHAPTER 6: MEASURING AND TESTING

6.1 Feedback

Whoever wants to improve something must know how it stands. A doctor measures fever, a school looks at its pupils' results, an agency counts how long an application takes. Software engineering has made this into an art of its own. A good program reports continuously on what it is doing, and a suite of tests runs with every change and reports when something has broken that worked before.

Such a test is worth only as much as what follows from its report. In software a strict rule therefore applies: an error that has been found is assessed at once and dealt with in a traceable way. Most are fixed immediately; the others are recorded, secured against and scheduled according to their severity, but none is left lying unattended. If errors are simply left standing, others are added, they mask one another, and after a few months nobody knows any longer which error triggers which report. A test that regularly raises an alarm and brings about nothing is worse than none, for it accustoms everyone to overlooking warnings. It must be repaired or removed.

6.2 When the Measure Becomes the Target

In institutions measuring takes a turn that it does not take in programs. A program does not know that it is being measured. People know it and act accordingly. If an agency is measured by its processing time, it deals with the easy cases first and leaves the hard ones lying. If a school is measured by its exam results, it practices the exam instead of the subject. If a hospital is measured by its case numbers, it treats more cases, not necessarily better. The economist Charles Goodhart made the observation in the 1970s about monetary policy, and in the

version that has spread it reads: When a measure becomes a target, it ceases to be a good measure.[1]

An old joke from the planned economy tells of a match factory whose work was measured in cubic meters of wood processed. It fulfilled the plan effortlessly by turning whole tree trunks into a single giant match. Whether it ever happened does not matter; the joke hits home because everyone knows similar cases. Still, one should not laugh too loudly. Where companies run their departments by metrics, the same thing happens in a subtler form: sales sells what is easy to sell, not what the customer needs, and customer service closes tickets instead of solving problems.

In addition, not everything that counts can be counted. How many applications a clerk processes can be counted; whether he listened to the applicant cannot. If only the countable is measured, only the countable is done. That is the danger of management by metrics, which over recent decades has migrated from companies into agencies, schools and hospitals: it puts the measurable in place of the end, and the end against which everything should be measured (Ch. 1) disappears behind it.

Behind this lies a more general inversion. Optimizing is right as long as it serves the end. Overdone or aimed at the wrong measure, it turns against it. Then efficiency becomes more important than effect, and one quickly gets done what never needed doing at all. The metric rises while the matter gets worse. Or the demand to make everything perfect paralyzes, until nothing gets finished any more. In all three cases it is not optimizing that is wrong, but that it has come loose from the end.

6.3 Measuring Without Steering

It does not follow that one should not measure, but that one should separate measuring from rewarding wherever possible. A figure on which nobody's salary and nobody's standing depends stays honest; one on which both depend gets shaped. Whoever wants to know how an institution is doing therefore measures several things at once, among them things that are hard to gloss over, and asks

1 Charles Goodhart, "Problems of Monetary Management: The U.K. Experience" (DE: „Problems of Monetary Management: The U.K. Experience"), in: *Papers in Monetary Economics*, vol. 1, Sydney: Reserve Bank of Australia 1975. The widespread version comes from Marilyn Strathern, "'Improving ratings': audit in the British University system" (DE: „‚Improving ratings': audit in the British University system"), *European Review* 5 (1997), pp. 305–321, here p. 308; she takes the name "Goodhart's law" (DE: „Goodhart's law") from Keith Hoskin.

those for whom the institution exists. And he looks at the figures together with those who supply them, so that he learns what lies behind them.

For tests within the institution, the rule from software applies. An audit, a quality report, an inspection makes sense only if what it finds leads to a reasoned decision: a remedy, closer observation, or the deliberate decision to bear a risk. A test that has found nothing for years is therefore not superfluous; perhaps it finds nothing because it exists. The grievance is not that a correction does not follow every finding, but that there is no connection between finding and decision that can be checked. Where tests take place only so that one can say testing has been done, what arises is what one might call inspection theater: much effort, no result, and the certainty of everyone involved that it is only about the paperwork. The same holds for documentation that is written only so that one can later prove one did nothing wrong (Ch. 8.2). A test whose findings lead to no decision should be abolished or changed so that they do.

Further reading: *Capitalism — An Introduction* (in preparation), Ch. 29.10 (management by metrics in the state), *Truth* (2025), Ch. 17 (when examining is obstructed in the name of truth)

Summary of Chapter 6

Whoever wants to improve must measure, and a test is worth as much as what follows from its report. Errors that have been found are assessed at once and dealt with in a traceable way; otherwise they pile up and mask one another, and a test whose findings lead to no decision accustoms everyone to overlooking. That a test finds nothing for a long time does not make it superfluous. But people know that they are being measured: when a measure becomes a target it no longer measures, and if only the countable is measured, only the countable is done. In general: optimizing that comes loose from the end turns against it. That is why one measures several things at once, separates measuring from rewarding, asks those for whom the institution exists, and demands that a checkable decision follow from every finding.

CHAPTER 7: ERRORS

7.1 Showing Errors

A program that notices an error and silently runs on as if nothing had happened is more dangerous than one that reports the error and responds to it in a controlled way. That is one of the first lessons of software engineering. Controlled does not always mean stopping; a program that flies an aircraft must not simply halt but must switch to a safe fallback mode and report that it has done so. Whoever swallows an error delivers a result that does not show it is wrong, and the damage appears much later at a point where nobody can find the cause any more. Good programs make their errors visible immediately and where they arise.

In institutions errors are swallowed because it pays. Whoever reports an error is asked who made it, and often it is the one reporting it. So one does not report it, fixes it quietly if possible, and hopes if not. The institution learns nothing from this, and the same error comes back, with someone else.

7.2 Looking Into It Instead of Looking for Culprits

In aviation and in some hospitals a different attitude has become established, and software engineering has adopted it. After an error one asks first not who is to blame but how it could come about: which procedures, handovers and assumptions made the error possible. The result is written down and made accessible to all, and it leads to a change that makes the same error less likely in future. Whoever reports an error is not punished for it, as long as he was not reckless or malicious.

This attitude is no license. It distinguishes between the error that anyone could have made because the procedure invited it, and the breach of duty that someone committed although he knew better. Often both are present at once. The overtired nurse who mixes something up has made an error that she could have avoided, and at the same time her duty roster invited it. A good investigation therefore determines the responsibility of the person and that of the institution

side by side, instead of playing one off against the other. Whoever holds a person responsible is not responsible alone, and whoever writes the duty rosters is answerable for the conditions under which others can or cannot fulfill their duty. Only this makes it possible to report errors without accusing oneself, and without the institution hiding behind the individual.

7.3 Where the Transfer Breaks

A program feels no shame. People have a reputation, a career and sometimes a family that lives off their job. An institution cannot take away their fear of reporting by a rule, only by experience: if the first who report an error are not punished but see that something changes, the next ones report too. If even one is punished, everyone learns the opposite.

It weighs more heavily that some institutions may not admit their errors, because their standing or their power depends on the appearance that they make none. An agency that withdraws a wrong decision admits that it was mistaken, and that can lead to lawsuits. A party that abandons a failed program hands its opponent a weapon. Where this is so, the institution as a whole swallows its errors, and no rule inside it can change that. Only an environment that punishes admitting an error less than concealing it can change it.

Further reading: *Thinking – Ordering Without Losing* (2025), Ch. 15 (orders that may not admit their errors), *Ethics – The Good We Already Do* (2026), Ch. 12 (institutions that preserve and lose reasons)

Summary of Chapter 7

An error that is swallowed does damage later, where no one finds the cause; that is why errors are shown immediately and where they arise. In institutions errors are swallowed because the one who reports often ends up looking like the culprit. What helps is an attitude that asks first how it could come about and changes the procedure, and that distinguishes the error the procedure invited from the breach of duty, without overlooking that often both are present at once. No rule takes away the fear of reporting, only the experience that reporting pays. Where an institution as a whole may not admit errors, only an environment helps that punishes concealment more severely than admission.

CHAPTER 8: VERSIONS AND TRANSITIONS

8.1 Do Not Overwrite

Whoever changes a text, a rule or a plan usually overwrites the old one. A year later nobody knows what applied before and why it was changed. Software engineering has solved this thoroughly differently: every change is recorded as a new version, with the name of the person who made it, the time and a sentence on the reason. Every earlier version can be restored, and for every line one can say when and why it came to be as it is.

The publishing house from Chapter 3 does the same with its manuscripts. A change is filed as a new version, the old one goes into the archive, and at the changed passage it says who made the change, when, why, and what stood there before. The publishing house consists essentially of one person and the programs that person works with; it knows no conflicts between departments. But even someone working alone forgets after half a year why he decided something the way he did, and with the recorded reason he can later check and correct his own decision.

8.2 The Memory of an Institution

In institutions knowledge leaves with the people who have it. The clerk who has known for twenty years why a particular case is handled the way it is retires, and her knowledge goes with her. What she leaves behind are files that say what was decided, but not why. Her successor sticks to what she finds, or she changes it without knowing the reason (Ch. 4.2).

An institution that writes down its reasons along the way has a memory that outlasts the people. It costs one sentence for each decision, a page for big ones. In return every rule and every decision can be traced back, including the wrong one, and whoever wants to change it later knows what he is changing. In a large

institution there is the further point that the reason serves not only memory but also the other person, who wants to check whether his knowledge was weighed or overridden (Ch. 5.3).

But writing things down has a reverse side that everyone knows who has worked in a public agency. There everything is often recorded and nothing is learned, because the recording serves another end: covering oneself. Whoever documents so as to be able to prove later that he acted according to regulations writes differently from someone who records so as to be able to check later whether he was right. The first writes a lot and says little, for every clear reason would be open to attack. The second writes little, but the reason that matters. How to tell them apart is simple: whether what was written down is ever read again and whether it then changes anything.

8.3 Transitions

Whoever migrates a program does not switch everything over in one day. He lets old and new run side by side for a while, checks whether the new does what the old did, and only then switches over. The same holds for institutions. A new law needs transitional rules for the cases that began under the old one, a new department a period in which the old one can still be reached, a new procedure a way back in case it does not work.

8.4 Where the Transfer Breaks

In a program the old version is just old text. In institutions old rules have people attached to them who made them, who do well with them or who have grown used to them. What in software is called legacy is often a vested interest in institutions. That does not make transitions impossible, but it makes them a question that cannot be answered only technically: Who loses through the change, who gains, and does one owe the losers compensation or only a reason?

Further reading: *Thinking – Ordering Without Losing* (2025), Ch. 15 (dead order)

Summary of Chapter 8

Whoever changes does not overwrite but creates a new version, with the name of the one who changes, the time and the reason. Then every rule can be traced back, including the wrong one, and the institution has a memory that outlasts the people who leave. Things are written down in order to learn, not to cover oneself; the touchstone is whether what was written down is read again and changes something. Switching over does not happen in one day but with a period in which old and new run side by side, and with a way back. But people are attached to old rules; what in a program is called legacy is often a vested interest in an institution, and a transition must say who loses and what is owed to them.

CHAPTER 9: RULES THAT A MACHINE EXECUTES

9.1 Like Cases Alike

Two people apply for the same thing, with the same documents, to two clerks in the same agency. One receives the benefit, the other does not. This is not ill will; the regulation leaves latitude, and each interprets it slightly differently. For the one who is refused it is nonetheless an injustice, for like cases are to be treated alike.

Where a rule is precise enough, a program can execute it, and it treats like data alike. It does not tire, has no favorites and no bad days, and one can look up which rule it decided by. That does not yet make it just: the rule itself can be unjust, the data can be wrong, and two cases can differ in something the program does not capture at all. Uniform application is not yet justified equal treatment. But where rule and data are correct, many simple cases can be dealt with faster and more evenly this way, and the people in the agency have time for the hard ones. In some countries tax assessments, child benefit or licenses are already granted this way for the ordinary case.

9.2 The Case the Rule Did Not Foresee

Every rule is written for cases its author had in mind. Others come along. The family whose income fits no column because the mother works abroad. The application that comes too late because the applicant was in hospital. A person in the agency can see that the case is different and decide according to the end of the rule. A program applies the rule, and the case falls through.

That is why every rule that a machine executes needs a path to a human being. Whoever thinks his case has not been captured correctly must be able to say so, and someone who is authorized to decide differently must examine it. This path must not be an obstacle course, or else it is open only on paper, and at its

end there must be a person who has the authority, the expertise and the time to change the machine's decision. And what is found along this path belongs back in the rule, so that the next like case does not fall through again (Ch. 6.1).

9.3 Where the Transfer Breaks

A program follows its rule even where it obviously misses its end. A person sometimes does that too, out of fear or habit, but he is capable of doing otherwise. Whoever hands decisions over to machines loses this possibility unless he explicitly builds it back in. In addition there is the question of who answers for a wrong decision. The clerk can say he made a mistake. Behind the program stand those who wrote the rule and those who translated it into code, and none of them feels responsible.

Weightiest of all is that a rule can become invisible in a program. Anyone can read and dispute a regulation on paper. A rule that exists only in the program is often known only to those who wrote it, and in programs that learn from examples instead of rules, nobody knows it exactly. An institution that decides according to such rules must therefore be able to say according to which considerations a decision was made, and in such a way that the person affected understands it and can dispute it.

Further reading: *Ethics – The Good We Already Do* (2026), Ch. 15.5 (what the rule did not foresee), *Truth* (2025), Ch. 19.6 (filtered without anyone censoring)

Summary of Chapter 9

Like cases are to be treated alike, and where rule and data are correct, a program applies the rule more uniformly than a person; people then have time for the hard cases. But uniform application is not yet justified equal treatment: the rule can be unjust, the data wrong. And every rule meets cases its author did not foresee. That is why every rule that a machine executes needs an open path to a person who is authorized to decide differently and has the expertise and time for it, and what that person finds belongs back in the rule. Whoever has decisions made by a machine must say who answers for the error and according to which considerations the decision was made, in such a way that the person affected can understand and dispute it.

CHAPTER 10: OPENNESS

10.1 Reasons One Can Read

A municipality decides to close a school. The parents learn of it from the newspaper, with one sentence by way of justification. Whether there were figures, which options were examined and who proposed what, they do not learn. They take the decision to be arbitrary, even if it was not, and they cannot check whether it was.

In software engineering it has become established for many programs that their source code is open. Anyone can look at how they work, find errors and propose improvements. This has proved itself because many eyes see more than few and because whoever knows that others are reading along writes more carefully. For institutions this means: reasons, budgets and procedures are open to view, as far as nothing speaks against it. Whoever is affected by a decision can read up on what it rests on.

10.2 Openness Binds

Openness is more than a service to the curious. It is the precondition of almost every control from outside, but not yet the control itself. An agency can publish all its reasons and still decide badly if nobody can object and enforce a correction. Accountability therefore includes three things: being able to know, being able to object, and being able to bring about a change; in the state through courts and independent oversight, in a club through the members' assembly, in a company through supervisory bodies, codetermination and law. Where these three come together, openness binds strongly. A management that must state its reasons before those whom its decision affects cannot hide behind forms (Ch. 5.4). An administration whose procedures are open to view cannot hoard its knowledge as an instrument of domination. What accountability an institution demands internally holds only once it gives account externally.

10.3 Where the Transfer Breaks

A program has no privacy. In institutions many files contain people: their illnesses, their debts, their family circumstances. Protecting this information is not a lack of openness but a duty toward those it is about. Deliberation too sometimes needs protection. Whoever knows that every sentence will be in the newspaper tomorrow no longer says what he thinks, and then it is precisely the dissenting voice that is lost which Chapter 5 wanted to record.

The rule therefore reads: open to view are the reasons, the procedures and the figures on which a decision rests. Protected are the data about individual people and, for a limited time, the frank word in deliberation. What is protected must be stated, and why. What makes the institution look bad is not concealed; that is precisely what openness is there for.

10.4 Who Controls Others

The institutions that are supposed to control others need control too: works councils and trade unions, supervisory boards, courts of audit, consumer associations. They ensure that a management must state its reasons, that employees are heard, and that the standards by which their work is judged are changed. Where an institution measures its staff by the number of tickets closed, a body that represents employees can do more than haggle over lower targets: it can ask what is to count as performance at all.

But the same institutions can make every error against which they were set up. A body can have so many meetings and committees that no time remains for the actual questions. It can measure its success by the number of its agreements instead of by what they bring about. It can reproach the management for lack of openness and itself decide without telling those it represents its reasons. And it can override a justified objection from its own ranks because the one who raises it does not belong to the majority.

That its members are elected does not by itself help against this. Election establishes who may represent; it does not guarantee that judgment is good. That is why the same questions apply to the controllers as to the controlled: Whom do their procedures serve, what do they learn from those they represent, what happens to dissenting voices, how do they notice their own errors? Their strength is that many together achieve what no one can achieve alone. This strength holds only as long as the common institution gives effect to its members' judgment instead of replacing it.

Further reading: *Hegel's Philosophy of Right – How Freedom Becomes Real* (2026), Ch. 29.8 (how a power binds itself), *Truth* (2025), Ch. 17.6 (when the state steers examining)

Summary of Chapter 10

Whoever is affected by a decision should be able to read up on what it rests on: the reasons, the options examined, the figures. Openness has proved itself in software because many eyes see more and whoever knows that others are reading along works more carefully. For institutions it is the precondition of almost every control from outside, but not yet the control itself: that requires being able to know, to object and to bring about a change. Accountability inside holds only once the institution also gives it outside. Protected are the data about individual people and, for a time, the frank word in deliberation, and for this too the reason is stated. What is not protected is whatever merely makes the institution look bad. Whoever controls must also let himself be controlled; that he is elected does not guarantee that he judges well.

CHAPTER 11: WORKING WITH LANGUAGE MODELS

11.1 What Has Become Cheap

Much of what this book proposes takes time: briefing papers with the rejected options, a second view on every important decision, recording the reasons, the check after a year, the good handover. Between people this quickly becomes an effort that stifles the work, and so it does not happen. With language models, programs that read and write texts, much of it costs almost nothing any more. A model can compile possible counter-reasons for every briefing paper, which must then be examined, turn a conversation into minutes that include the reasons, and after a year compare the assumptions made back then with what actually happened.

What has become cheap, however, is writing, not examining. A model that knows only the briefing paper can find the usual counter-reasons; what is missing from the paper because nobody thought of it, it cannot know. Whether its counter-reasons are the decisive ones must still be judged by a person who knows the matter. And where patient records, personnel files or confidential matters are involved, using a model is not a question of convenience but of protecting those they are about, and often a question of law.

Within these limits the question of effort that Chapter 12 raises shifts. What used to be reserved for the big decisions can now be afforded for middling ones too, and an institution that does not use this gives away an opportunity to become better.

11.2 What a Model Can and Cannot Do

An old saying from the time when the first computers came into offices lists the ways to ruin a company and names the computer as the most thorough. Another, from the same time, says the same thing more briefly: garbage in, garbage out.

Both hold for language models in a new form. A model writes a convincing briefing paper from false information, and it does not know what is missing from its material. What it delivers always looks finished, even where it is not.

A model does not have the interests of an employee: it does not want to be promoted, protects no department and fears no superior. Many of the troublesome peculiarities of human staff are absent in it, but it has others. It tends to report things as done when they have been dealt with, to summarize more smoothly than it has checked, and to agree readily with whoever asks it. For working with it, therefore, the same holds as what Chapter 5 demands of human preparers: have uncertainty stated, test the result independently, keep the way back open. Whoever neglects this does not ruin his institution faster than before, but more thoroughly.

11.3 Two Dangers That Reach Further

A model carries out the ends it is given, and it does so more thoroughly than people. An employee is not wholly absorbed in his role. He is also a father or mother, a neighbor, a citizen, a member of a congregation or a club, and from there he sometimes objects where the end of the institution violates what he otherwise holds to be right. This correction from a person's other roles is one of the most important safeguards institutions have, and hardly any of it is prescribed anywhere. Where models do the work, it falls away. The danger is therefore less that machines take over power against the will of human beings than that they realize the prevailing ends too well, at points where people would still have corrected them.

The second danger is quieter. A model can keep a broken institution alive. It writes the briefing papers that nobody writes honestly any more, summarizes the reports that nobody reads, and compensates for the shortcomings of the handovers, so that nobody notices any longer how badly the institution is built. What would earlier have failed because of its shortcomings, and thereby forced a reform, now carries on. The model's help spares the institution reform at the root.

Neither speaks against using language models. Both speak for every institution laying down where a person decides and can object, and for treating whatever the model compensates for as a finding: as a pointer to a place that needs to be repaired.

Further reading: *Ethics – The Good We Already Do* (2026), Ch. 15.8 (knowledge of consequences), *Truth* (2025), Ch. 19 (calibration)

Summary of Chapter 11

With language models much has become cheap that was too expensive between people: possible counter-reasons for every briefing paper, recorded reasons, the check after a year. But what has become cheap is writing, not examining; whether the counter-reasons are the decisive ones must be judged by a person who knows the matter, and where information about people is concerned, protection and law set limits. Within these limits one should use it. A model has no interests like an employee's, but biases of its own: from false information it makes convincing texts, it reports things as done, and it agrees with whoever asks. That is why what holds for every briefing paper holds for its work: state uncertainty, test independently, keep the way back open. Two dangers reach further: a model carries out the given ends without the objection that people bring from their other roles, and it can keep alive a broken institution that would otherwise have been reformed. Every institution therefore lays down where a person decides, and treats whatever the model compensates for as a finding.

CHAPTER 12: HOW MUCH EFFORT

12.1 Proportionate

Every form this book proposes costs something. If all of them are demanded everywhere, the result is the very bureaucracy they were meant to help against: briefing papers with counter-reasons for the purchase of a printer, error reports for every typo, a transitional rule for moving the coffee machine. The question of how a department makes its coffee needs no dissenting vote in the minutes; for the question of whether a town loses its maternity ward, it is the least that is required.

The effort depends on two questions: How much depends on the decision, and how hard is it to reverse? What affects many and cannot be undone is prepared, examined and justified thoroughly. What affects few and can be done differently tomorrow is decided. In small matters a decision that is not quite right is often better than none; whoever weighs things up until the occasion has passed has also decided.

In institutions hesitation takes a form of its own. A decision is examined until it is secured against every conceivable objection, and that usually means: never. The safeguarding then protects not those affected but those who were supposed to decide.

12.2 Examining One's Own Rules

An institution must put this question to its own rules, and again and again. What is this step for, and is it worth the effort? Rules are seldom abolished and often added, and each one has a good reason. Only their sum stifles the work. Conversely, an institution that constantly introduces new things is not thereby capable of learning; it can switch methods without ever asking why the old ones failed. An institution is capable of learning if it decides for reasons what it preserves, what it changes and what it gives up. That is why it is not enough to justify each new rule; from time to time one must measure the old ones against their

end and strike out those that no longer fulfill it or fulfill it with too much effort (Ch. 4.2).

A program cannot do this. It executes what it has been told to do and never asks whether the instruction still makes sense. People can, and that is the difference that matters. An institution whose people may and should measure its rules against their end is a learning organization, even if it works strictly by rules. One in which they only execute is bureaucratic in the bad sense, even if it consists entirely of clever people. A good public agency is not meant to be without rules, but binding, reliable and capable of learning at once.

12.3 The Machine Made of People

It does not follow that everything mechanical in an institution is an evil. A body contains mechanics and chemistry: bones that bear loads like beams, joints that move like hinges, a metabolism that runs as in a laboratory. That is not a defect of the living but its basis. It happens, however, within a whole that has an end of its own and that everything mechanical serves. The mechanics become noticeable only when something breaks: the bone, the joint, the metabolism in illness.

In an organization it is the same. Much has to run as if by itself: accounting, filing, cover during holidays. Whoever had to rethink each of these routines every day would get nothing done, and that they run mechanically frees people for what demands judgment. An institution becomes a bureaucracy not because it contains mechanical elements, but because the mechanical no longer serves an end that the people in it share and can examine. Then the routines run for their own sake, and the people become their parts. They do what the routine demands; their judgment, their other roles and their objection no longer have a place in it.

According to the American historian Lewis Mumford, this machine made of people is older than any machine made of metal. The first great empires built their pyramids, canals and armies by arranging thousands of people like the parts of a mechanism, and Mumford called this the megamachine.[1] What it achieved was immense, and immense was what it cost those of whom it was made. Software engineering did not invent this danger. It can sharpen it where

1 Lewis Mumford, *The Myth of the Machine*, vol. 1: *Technics and Human Development* (1967), vol. 2: *The Pentagon of Power* (1970); German: *Mythos der Maschine. Kultur, Technik und Macht*, Vienna: Europa-Verlag 1974.

programs set the pace at which people work, and it can help to banish it where it takes over the mechanical and leaves judgment to people.

12.4 When the Rules Contradict One Another

The rules of this book come into opposition with one another as soon as each is applied on its own without restriction. That is no oversight; it shows that each holds only under certain conditions. One thing, one place (Ch. 3) does not go together with independent control. The simple rule (Ch. 4) does not go together with the individual case (Ch. 9). Measuring (Ch. 6) does not always go together with the end, openness (Ch. 10) not with protection.

A ranking of the rules does not help here, for it would have to provide an exception for every case. What decides is the same thing with which the book began: the end of the institution and the question of whether its reasons hold up before those whom a decision affects. What is sought is not a compromise in which each rule gives way a little, but a more precise determination of when which one holds: the authoritative source for daily work, the independent version for control; openness for the reasons, protection for the people. Whoever faces the contradiction of two rules asks which of the two serves the end here, and whether he could convince, with his reason, the person whom the other rule was meant to protect. That is no formula but an examination, and it demands judgment. It is precisely this judgment that distinguishes an institution from a program.

Further reading: *Ethics – The Good We Already Do* (2026), Ch. 6 (justification before those affected), *Ethics – The Good We Already Do*, Ch. 15.7 (too much and too little), *Thinking – Ordering Without Losing* (2025), Ch. 15 (dead order), *Hegel's Philosophy of Nature – The Idea in Its Otherness* (2026), Ch. 21 (mechanism, chemism and the living)

Summary of Chapter 12

Every form costs something, and demanded everywhere it produces the bureaucracy it was meant to help against. The effort depends on how much depends on a decision and how hard it is to reverse; in small matters a decision that is not quite right is often better than none. An institution must measure its own rules against their end again and again and strike out whatever no longer fulfills it. No program

can carry out this self-examination, and it distinguishes a learning organization from one whose rules have become independent. Mechanical elements belong in every institution as bones and metabolism belong in the body; it becomes a bureaucracy only when the mechanical no longer serves an end that the people in it share, and the people become parts of a machine. Where the rules of the book come into opposition because each is applied without restriction, no ranking decides, but the end of the institution and the question of whether its reasons hold up before those affected.

CHAPTER 13: WHY REASON HAS A CHANCE

13.1 Making the Right Thing Easy

One can demand of people that they do the right thing against their own advantage. Sometimes they do. But an institution that relies on it will be disappointed, and it wears out those who do it. Better is an institution in which the right thing is easy and the harmful thing hard: in which the honest briefing paper is heard, reporting an error pays, the superfluous rule can be struck out, and the person affected learns why the decision was made as it was.

Built in this way, what is reasonable in an institution prevails not against people's interests but with them. Whoever is heard writes more honestly; whoever is not punished reports; whoever sees that his rule is being examined defends it with reasons instead of with power. How much depends on this is shown by a comparison often drawn between the market and the planned economy. What is usually meant is this: where one's own well-being depends on the institution doing its job well, one has a reason of one's own to consider, beyond the regulations, how things could go better. Where, on the other hand, one is measured above all by whether one has kept to the prescribed steps, thinking about their point often does not pay, and whoever does more gets little from it but trouble. Both occur in companies as in public agencies. There are companies in which all that counts is making no mistakes and reaching the metric, and agencies in which professional pride, responsibility toward citizens and the respect of colleagues carry much further than any service instruction. This lies less in the people than in the way the institution is built. Admittedly, the interest that a company arouses is not by that fact the reasonable either: it is directed at the survival and the profit of the company, not necessarily at its end for others. A good design connects people's own interest with the end of the institution, not only with its survival and not only with its regulations.

On this rests the hope of this book, and it is not new.

13.2 Hegel's Hope

Georg Wilhelm Friedrich Hegel formulated it for society as a whole. In the economy, he writes, everyone seeks his own, and yet "subjective selfishness" (DE: „subjektive Selbstsucht") turns into a contribution to the satisfaction of the needs of all others.[1] For the state he hoped for something similar: whoever experiences that the public institutions protect his trade and his rights knows the state as his own, and in this lies "the secret of the patriotism of the citizens" (DE: „das Geheimnis des Patriotismus der Bürger").[2] According to Hegel, reason works not only against interests but through them, where institutions are rightly built.

13.3 Whose Advantage

In 1843 the young Karl Marx objected to Hegel that the officials from whom Hegel expected concern for the universal make the end of the state into their private end, "a chasing after higher posts, the making of a career" (DE: „zu einem Jagen nach höheren Posten, zu einem Machen von Karriere").[3] The task does not guarantee its fulfillment. The objection applies to this book too. That the reasonable can be advantageous does not mean that it is advantageous, and certainly not for everyone.

That is why one must always ask whose advantage is meant and by what it is measured. An institution can be built so that its advantage is that of its management, or of its funders, or of its metrics, and then it is not the reasonable that prevails but the advantageous, and the two fall apart. That is the inversion that Chapter 6 showed in optimizing, on a large scale: what was meant to serve the end becomes an end itself. Where an institution stands under a standard that it has not given itself, for instance the pressure to compete with others for money, no good design inside it is enough; then the standard itself must be examined. An institution can be reasonably built within its task without the conditions in which it works being reasonable, and some institutions are very skilled at realizing an end that does not hold up before those it affects. Not every undesirable development is therefore the result of bad management.

1 G. W. F. Hegel, *Grundlinien der Philosophie des Rechts* (1820), § 199, TWA 7, p. 353.

2 Ibid., § 289 Remark, TWA 7, p. 458.

3 Karl Marx, *Zur Kritik der Hegelschen Rechtsphilosophie. Kritik des Hegelschen Staatsrechts* (1843), MEW 1, p. 249.

What a good design can achieve is nevertheless a great deal. It makes visible where advantage and reason fall apart, and it gives a voice to those who notice it: to the experts whose briefing paper is read, to the staff who report an error, to those affected who receive a justification. That is not the victory of reason, but it is its chance, and the chance is greater than the complaints about institutions would suggest.

Further reading: *Hegel's Philosophy of Right – How Freedom Becomes Real* (2026), Ch. 27.2 (the system of needs), *Hegel's Philosophy of Right – How Freedom Becomes Real*, Ch. 29.4 (the executive power and Marx's critique), *Capitalism — An Introduction* (in preparation), Ch. 25 (how an order holds through what people do), *Ethics – The Good We Already Do* (2026), Ch. 15.8 (making the right thing easy and the harmful thing hard)

Summary of Chapter 13

An institution that relies on people doing the right thing against their own advantage will be disappointed. Better is one in which the right thing is easy and the harmful thing hard; then what is reasonable prevails with people's interests instead of against them. That was Hegel's hope for the economy and the state. Marx objected to him that the task does not guarantee its fulfillment, and the objection applies to this book too: one must always ask whose advantage is meant and by what it is measured, and where a standard rules from outside, it must itself be examined. A good design makes visible where advantage and reason fall apart, and gives a voice to those who notice it. That is not the victory of reason, but it is its chance.

CONCLUSION: WHERE ONE CAN BEGIN

One Process Instead of a Reform

Whoever has read this book and thinks of his own institution will be tempted to want to reorganize everything at once. That almost always fails, and it contradicts what Chapter 4 says about growing from simple beginnings. It is better to begin with a single process that is going badly and to follow it as it really runs.

A club needs six weeks to approve an event. Instead of changing the statutes, one takes the last application and traces its path: which units had it in hand, which information was requested more than once, where it was left lying, which check was meant to prevent which damage, and who decided in the end. After that one usually knows what has to be changed, and often it is less than one thought.

Seven Questions

For such a process the course of this book can be summed up in seven questions; they are not exhaustive. They are not a method that replaces judgment, but a reminder of what judgment should ask about.

1. What is the procedure for, and does this end hold up before those it affects?
2. How does it really run, not only according to the regulations?
3. Where is knowledge lost, work done twice, or a decision left lying?
4. What problem was the rule that gets in the way meant to solve originally?
5. Who gains and who loses if one changes it?
6. Who may decide on the change, and who must be heard?
7. By what will one recognize after a year whether things have got better?

What One Can Do Where One Stands

Not everyone can change what he has recognized as wrong, but everyone can do something. Around every person in an institution lies a circle of what he has influence on, and a wider one of what concerns him without his being able to influence it. Whoever directs his strength only at the outer circle exhausts himself. Whoever begins with the inner one usually finds that it is larger than he thought and that it grows with time. But the boundary between the two is not fixed. An old formula asks for the strength to change what one can change, the patience to bear what one cannot change, and the wisdom to tell the two apart. It knows two levels and overlooks a third that lies between them: what one person alone cannot change and many together can. That is astonishing, for almost everything people do they do together, from the family through the club to the company, and this whole book is about how such joint work is organized. Some change begins when colleagues, members or citizens notice that they have the same problem, and it goes beyond the institution where necessary: into trade unions and works councils, into citizens' initiatives, into elections, where many together achieve what no one achieves alone.

Even someone who remains alone has an effect beyond his circle. How work is done in an institution, whether one may ask about the point of things, whether errors are reported, whether people tell one another the truth, is laid down in no regulation. It is its culture, its spirit, and that consists of what its people do day after day. Everyone contributes to it, whether he wants to or not, and what someone exemplifies can change what seems self-evident to the others. Whether it does so depends on the institution too; where objection is punished, setting an example costs a great deal and achieves little on its own, and there the others are needed. The advice to begin with oneself is therefore no call to accept the rest.

Even the most ordinary post has latitude, for no regulation governs everything. A clerk in a public agency is bound by laws, responsibilities and official channels and may not override a regulation, even if he no longer sees its point. But he can try to understand what it is for and, where it leaves latitude, apply it according to its end rather than its letter. He can hide behind it or tell the citizen openly what applies, why, and what the citizen can do. He can record what consequences a regulation has, in cases that others can check, and propose a change through the channel provided for that. The same holds for the employee in a company toward her customers and colleagues and for the club member toward the board: give feedback, ask about the point, tell the truth.

The higher someone stands in the hierarchy, the greater, as a rule, is his authority to decide. That is not the only power in an institution; a workforce that

acts together can move more than a single head of department, and a managing director can be tightly bound by her owners. A head of department can largely organize her area herself, as long as it meets its targets: how things are handed over, what briefing papers look like, whether errors can be reported without anyone being punished. She can carry feedback upward to where it will be heard again. A club chairwoman can have a superfluous rule abolished by resolution the very next evening. A managing director can strike out an internal procedure, but must check whether it fulfills a duty the law imposes on him.

Whoever heads a company, a ministry or a government can do the most, and faces a temptation the others do not: to introduce the latest management method of the day in the hope that it will solve everything at once. None does. More fruitful is an attempt that is seldom made: to get clear about what the institution is for, which goals follow from that, which means are at its disposal, and how these three work together. What otherwise prevails in institutions of its own accord, the requirements of the currently dominant method, the metrics, habit, would then be replaced by something more reasonable. Whether that succeeds is open. It is an attempt worth making. Someone who stands so high up is, moreover, answerable not only for his own decisions but for the conditions under which everyone else in the institution decides: what knowledge they receive, what they are measured by, whether they may object. He owes them the possibility of living up to their own responsibility.

Most of the rules that paralyze an institution have never been disputed, because nobody wrote down their consequences and nobody asked about their end. Whoever does so, at whatever point, gives those who may make changes a reason and those affected a voice. This book asks no more of anyone; it only shows that even this can change an institution.

Further reading: *Capitalism — An Introduction* (in preparation), Ch. 11 and *Capitalism — An Introduction*, Ch. 35 (what can only be changed together), *Ethics – The Good We Already Do* (2026), Ch. 15.8 (making the right thing easy and the harmful thing hard)

Whether it does so is the test by which this book must let itself be measured: not by how many read it, but by what has changed afterward in its readers' institutions.