

CHAPTER 1

Three Roles Beyond the Technology

Most AI projects fail over a question nobody asks: who decides, and when. Three roles — who trains, who controls, who is accountable — form the social layer of an AI system. Where they stay unfilled, or collapse into one person, the project fails on the human side of the architecture while the technology keeps running.

A I projects rarely fail at the model; the model usually works. They fail because, after go-live, the system falls into a gap where nobody is responsible any more. The data scientist has moved on to the next pipeline. The engineer who keeps the models in operation watches monitoring dashboards whose alert thresholds he set himself. The plant manager knows that a model is running somewhere — and that is fine; he owes nobody an intimacy with the training data or with the supplier of the GPU layer, the specialized processors the model computes on. The problem sits elsewhere: when the model gives a wrong recommendation in an edge case — and that happens, every month, in some shift — the organization looks for the person with the mandate to decide. It finds no one.

What looks like a software problem is an anthropology problem. Three roles were never spoken out loud before the system went into operation. Three responsibilities were never assigned, three escalation paths never built: who trains, who controls, who is accountable.

What anthropology means in a technical system

Anthropology sounds academic. Here the word carries a narrow, applied meaning: it asks which people, holding which mandates, are indispensable at which points of an AI system. What matters is less the skill of building the model — that is one part of the system and the duty of a narrower role — than the position in the architecture that keeps the model working over the long run.

An AI system consists of two layers. The technical layer holds code, data, model weights and infrastructure. The social layer holds people with clear roles — without them the technical layer starts, within a short time, to drift, to fail, or to produce a risk that nobody controls any more.

The distinction is older than the current generation of models. Norbert Wiener, in *The Human Use of Human Beings*, already insists that an automatic system does not exist without a human control loop in which its readings are interpreted and its behavior corrected.[¹] Stuart Russell, in *Human Compatible*, carries the same intuition into the present frame: without an explicit holder of the control mandate, an autonomous system by definition operates outside the loop.[²]

Whoever builds an AI system without constructing the social layer has bought expensive computation — processors, models, infrastructure — without building the escalation paths, so at the first incident nobody is positioned to step in. That has nothing to do with fine-tuning the model and everything to do with organization.

Three roles that appear in no model documentation

Every productive AI pipeline — from training and deployment through daily operation — carries three roles. They are distinguishable, they must never collapse into one person, and they appear in no organization chart automatically.

Who trains. This person determines what the model will take for reality. She approves the training data, accepts or filters out outliers — single records that deviate so strongly from the rest that they can distort the model — defines the labeling scheme and decides which assumptions get encoded explicitly. Whoever trains stamps the reality in which the system operates. *Data curator* names the role better than *data scientist*, because it captures the substance of the work: selecting, filtering and documenting what enters the training set — and therefore becomes reality for the model.

In most industrial projects this position is invisible. Training data gets pulled from a report in the company's resource-planning software without anyone checking whether a strike day distorts the average behavior. A dataset from 2022 flows into the model although the line was rebuilt in 2025. Nobody objects, because nobody holds the mandate to object.

Who controls. This person holds the mandate to intervene in live operation. She defines the thresholds at which the model is put into question, and the escalation paths that get used when it matters. She holds override rights — the documented right to overrule a system decision and replace it with her own — and she logs every use of them.

What she does not do: trust the model blindly. What she equally does not do: bypass the model and decide by gut feeling because that is quicker and more familiar. She is the *30-percent layer* — the human share of a 70/30 design, the share that keeps edge cases from becoming escalations and escalations from becoming standstills.

Who is accountable. This person carries the legal responsibility. Under the EU AI Act that means: she signs the declaration of conformity, she maintains the technical file, she reports serious incidents. The role must be assigned in writing, with insurance coverage and a clear mandate. It does not land automatically on the CTO, the plant manager or the compliance manager; it is a separate appointment.

Whoever is accountable must understand what the model does without having built it personally. The position demands a work of translation between technology and organization that the classical IT organization chart rarely provides.

What happens when all three roles fall on one person

The most frequent case in DACH industrial projects: a single plant manager carries all three roles informally. He signed off the project, knows the training data, authorizes interventions and signs the reports for the auditor. As long as this plant manager stays in office, the system runs. The moment he moves on, the project breaks — anthropologically, while the technology keeps humming.

One person cannot curate data, take mandate decisions and carry legal liability at the same time; the roles collide at every serious question. No plant manager who discovers a model error will press for his own liability — far more often the incident gets smoothed over, reclassified as a technical glitch, or parked under *we will look into it later*. Where an escalation should stand, silence grows, and silence is the most expensive variant.

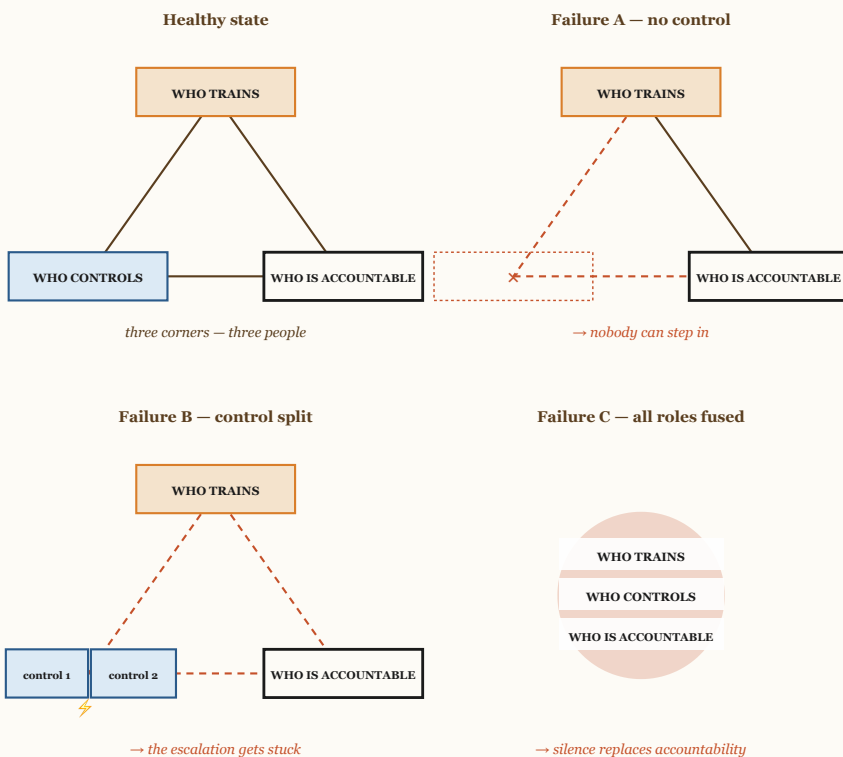
Why the roles must be explicit — an analogy

The programming language Rust has established a principle in the software world that serves here as an analogy. Its compiler — the checking program that translates human-written code into instructions a machine can execute, and refuses when the code is unsound — demands that every variable have a spoken role: who owns this piece of memory, who borrows it temporarily, which reference may change it and which may only read. Without these explicit assignments the program does not get built at all.

At first this looked pedantic. It turned out to be the precondition for reliability in production systems. What used to pass as *an experienced programmer will get this right anyway* is today a written contract structure between the code and its checking program.

The anthropology of an AI system follows the same logic. Who trains, who controls, who is accountable — these three assignments are the organizational counterpart of those annotations. When they are missing, the system goes into production anyway: the model trains, the pipeline runs. Eighteen months later the first escalation arrives, and nobody holds the mandate. The organization reaches for a responsibility that nobody holds any more — the AI equivalent of what programmers call use-after-free: code that reaches into memory already released, formally still executing, working on data that belong to nobody, until it fails unpredictably.

THREE ROLES — THREE DEFORMATIONS OF THE TRIANGLE



The healthy triangle has three occupied corners. Each failure is a typical structural deformation, not an individual weakness.

Three Questions for Your Own Context

Three questions to be answered operationally — on a sheet of paper with names and a date, without a strategy workshop.

1. **Who trains?** Which person, named by name, holds the mandate to accept, reject or curate training data? When was the role last confirmed in writing?

2. **Who controls?** Who holds override rights in live operation? Are thresholds and escalation paths documented, and who logs their use?
3. **Who is accountable?** Who signs the declaration of conformity under the EU AI Act? Who carries the insurance? Who reports incidents to the market surveillance authority?

If one of these questions has no name behind it — or the same name three times — the system is anthropologically unfinished.

Sources

1. Norbert Wiener, *The Human Use of Human Beings: Cybernetics and Society*, Houghton Mifflin, 1950.
2. Stuart Russell, *Human Compatible: Artificial Intelligence and the Problem of Control*, Viking, 2019.