

MOST MANAGEMENT BOARDS DO NOT UNDERSTAND THEIR OWN COMPANIES

The company as a network of dependencies

A company is not a collection of departments.
It is one operating system built from thousands of dependencies.

PEOPLE • MACHINES • INFORMATION • MATERIALS • WORKAROUNDS

MATEUSZ CHRZANOWSKI

SYSTEM OF SYSTEMS ARCHITECT

OPERATING REALITY / DEPENDENCIES / CONSEQUENCES



Most Management Boards Do Not Understand Their Own Companies

Most management boards do not understand how their own companies actually work. They understand financial results, KPIs, reporting lines and organizational charts. They understand the company as it appears in presentations. That is not the company.

An organizational chart shows hierarchy. It does not show who corrects whose mistakes, where information dies, which procedures are routinely ignored, how materials actually move or which employees keep the business running through unofficial workarounds. A company is not a collection of departments. It is one operating system built from thousands of dependencies. That is what I reconstruct.

REPRESENTATION

FINANCIAL RESULTS

KPIs

REPORTING LINES

ORG CHARTS

OPERATING SYSTEM

PEOPLE

MACHINES

INFORMATION

WORKAROUNDS

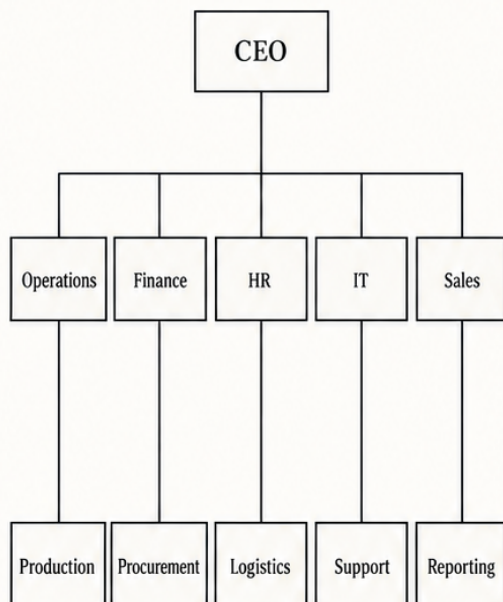
DEPENDENCIES

THE COMPANY ON PAPER vs THE COMPANY IN REALITY

SAME PEOPLE. MORE CONNECTIONS. A MORE COMPLEX REALITY.

THE COMPANY ON PAPER

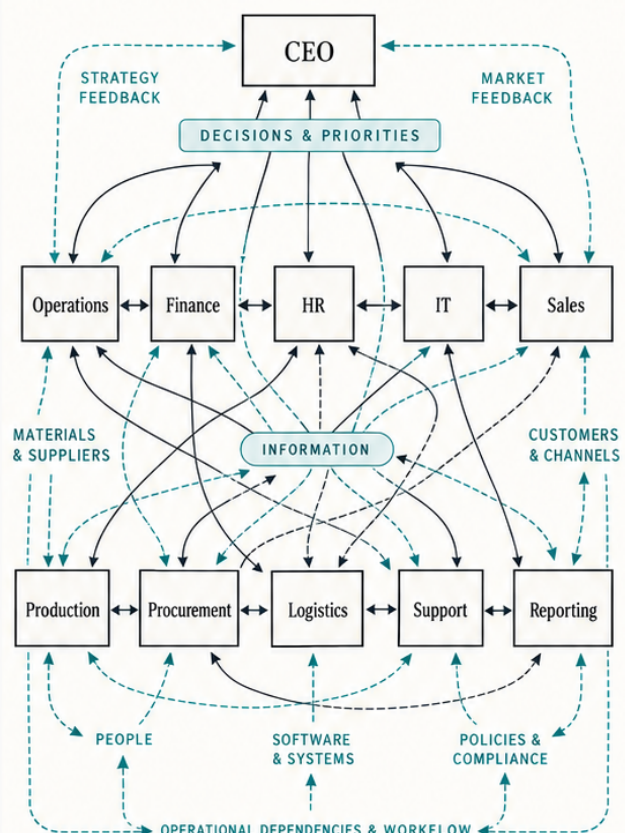
A CLEAN STRUCTURE.
CLEAR LINES. SIMPLE STORY.



HIERARCHY SHOWS
WHO REPORTS TO WHOM.

THE COMPANY IN REALITY

A NETWORK OF DEPENDENCIES.
MULTIPLE FLOWS. CONSTANT FEEDBACK.



REALITY INVOLVES PEOPLE, PROCESSES,
SYSTEMS AND EXTERNAL FORCES
ALL WORKING TOGETHER.

The chart is useful. The operating company is a network of dependencies.

I Do Not Begin With Management

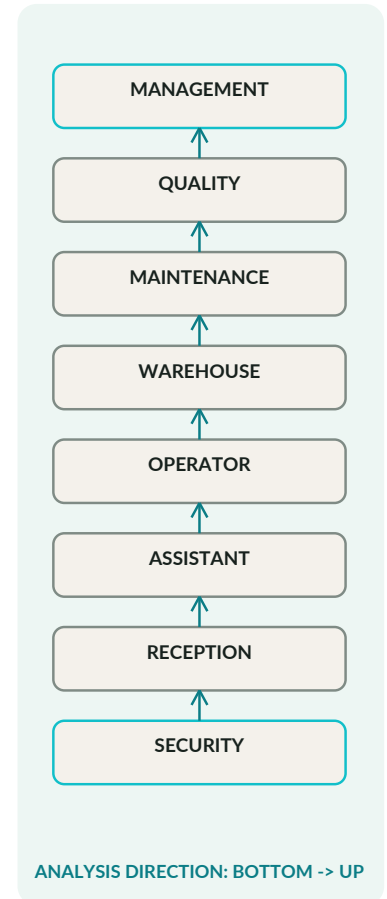
Management gives me access, direction and contractual authority. Then I leave management behind.

I do not begin by asking executives to explain the company to me. Their version would immediately frame the analysis through the same mental model I am there to examine. I am not interested in how the organization is supposed to work. I need to see how it actually works.

So I begin at the bottom: with the security guard, receptionist, secretary, machine operator, warehouse worker, driver, maintenance technician, quality inspector and production employee. Then I move through the organization, layer by layer.

No individual employee understands the entire company, but every employee knows something management usually does not. The security guard sees movement. The receptionist sees communication bottlenecks. The executive assistant sees how decisions really circulate. The operator knows which procedures are useless. The technician knows which failures keep returning. The warehouse worker knows where planning stops matching reality.

The company becomes visible only when those fragments are connected.



I Need Direction, Not a Guided Tour

I do not work alone. My assistant accompanies me and supports the collection and organization of material. When neither of us speaks the client's language, someone from the company must accompany us and speak fluent English. Preferably, that person is a secretary, assistant or someone from a lower organizational level.

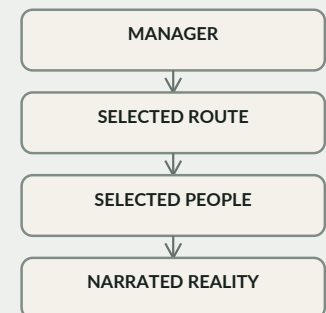
Their role is to make communication possible, not to interpret the company for me. I need to know where to go. I do not need a manager narrating reality, selecting what should be shown or explaining what I am supposed to see.

The visible process is usually easy to understand. The real architecture appears in conversation. I speak with the people performing the work, not with one carefully selected representative expected to speak for an entire group. If ten operators work on ten similar machines, I may speak with all ten because they do not experience the same system.

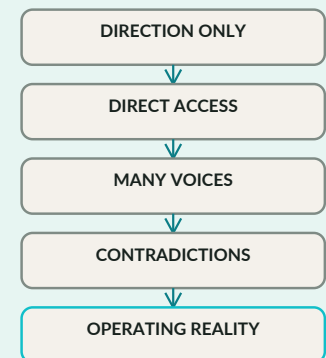
One knows why the machine stops. Another knows which material causes problems. Another knows that the formal procedure is ignored because following it would make production impossible. Another knows which defect appears two stages later. This is not inconsistency. It is the organization as it actually exists.

When two people describe the same process differently, both accounts may be true. The company may not have one process at all. It may have several local versions operating under the same name. Documentation records the official fiction. People reveal the operating reality.

GUIDED TOUR



ARCHITECTURAL FIELDWORK



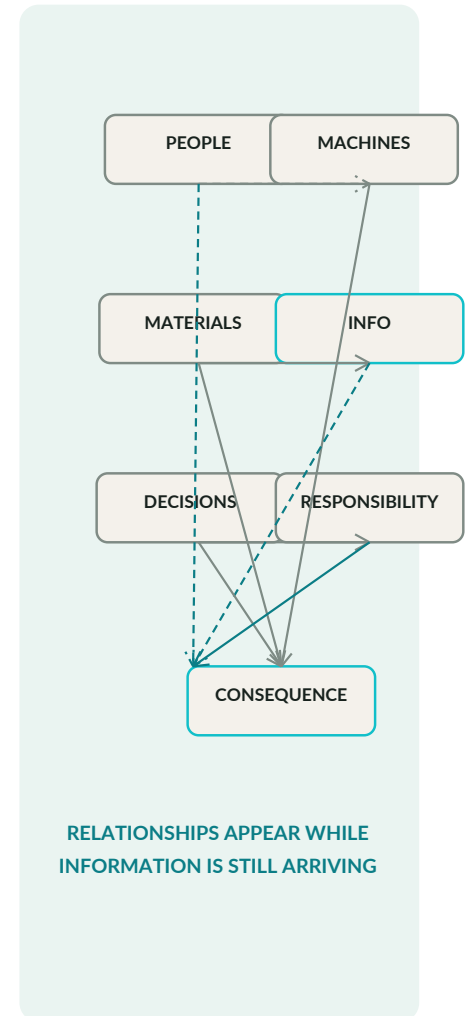
I Work in Graphs

I do not write notes. I draw.

From the first conversation to the final architecture, my working language is the graph. I do not collect linear notes and later convert them into a model. The graph itself is the architecture in development.

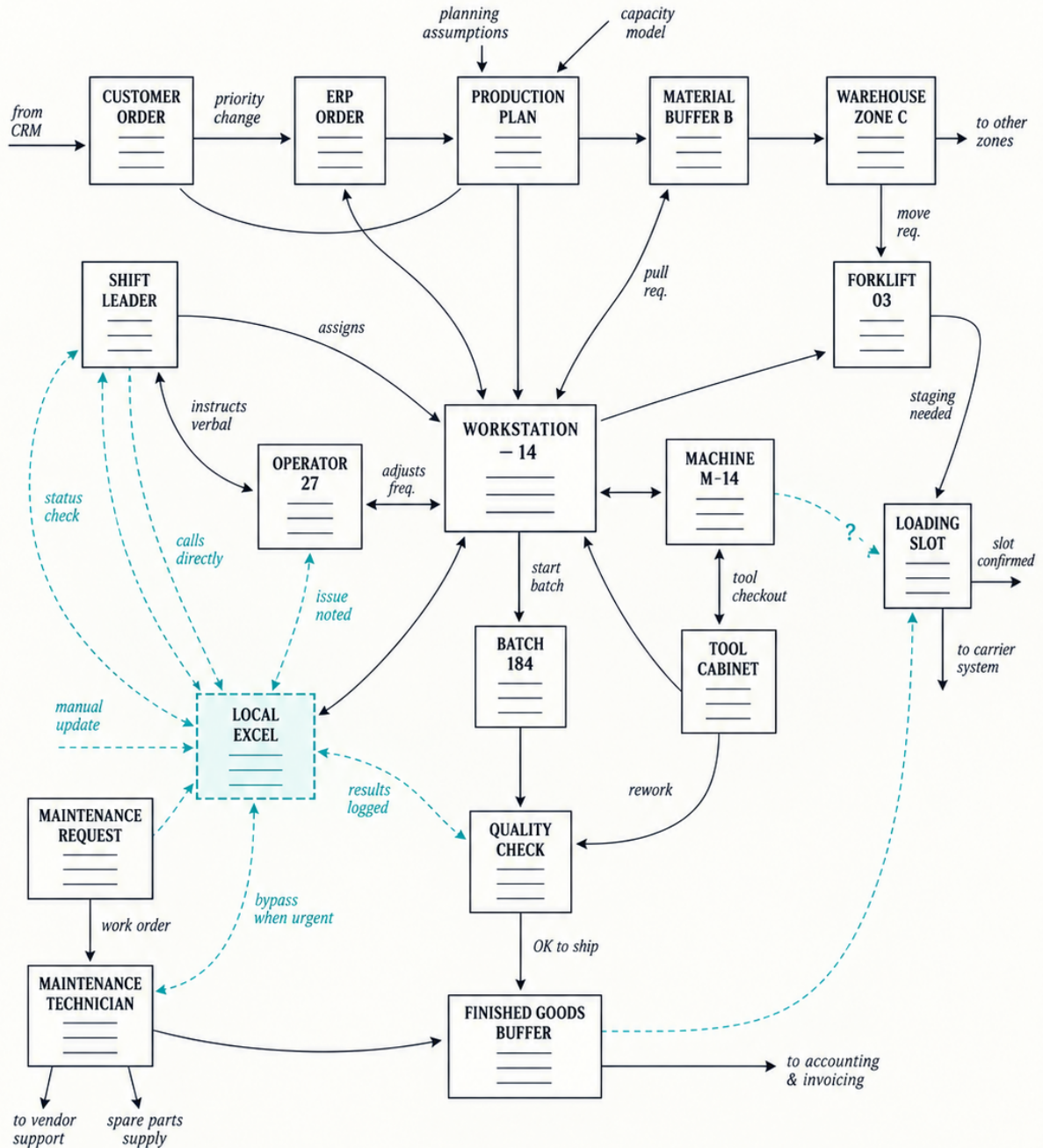
During field work, I use an iPad with an Apple Pencil, carried in a case suspended from a strap around my neck. It allows me to move through the organization while continuously expanding the model in real time. Back in the office, I continue the work on an Artist 22R Pro pen display, where the graph can grow into a much larger and more detailed structure.

This is not a matter of convenience. It is how I think. I see people, machines, information, responsibility, materials, decisions and consequences as connected layers. Most analysts collect information first and search for relationships later. I see the relationships while the information is still arriving. I walk into a production hall, see the process and see its dependencies at the same time.



THE GRAPH IS THE ANALYSIS

A FRAGMENT OF A HYPOTHETICAL OPERATING MAP



Fragment of a hypothetical operating map

PEOPLE. PROCESSES. SYSTEMS. A CLEARER PICTURE.

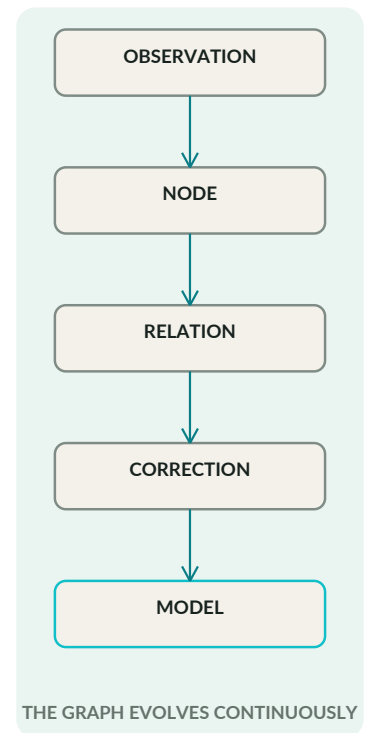
I Work in Graphs - continued

The graph evolves continuously throughout the engagement. Relationships are added, corrected and reorganized as new information appears. There is no separate phase in which rough notes are transformed into architecture because there are no rough notes. The architecture exists from the beginning, although it remains incomplete until the system has been fully reconstructed.

I use photographs only when I need to attach a precise observation to a physical object: where a machine jams, why a workstation is poorly designed or which condition creates unnecessary work. The photograph supports the graph. It does not replace it.

For a medium-sized or larger company operating from one location, the mapping itself usually takes about three weeks. If the engagement covers mapping alone, the completed model of the existing organization becomes the final product for the client. If I am also designing a new architecture, I continue working on the graph until that architecture is complete.

Only then does my graphic designer prepare the finished model for delivery. The designer does not analyze the company, interpret the findings or create the architecture. The designer converts my completed working model into a professionally structured client version that is clear, consistent and readable to someone other than me.



The Company Comes Before Its Processes

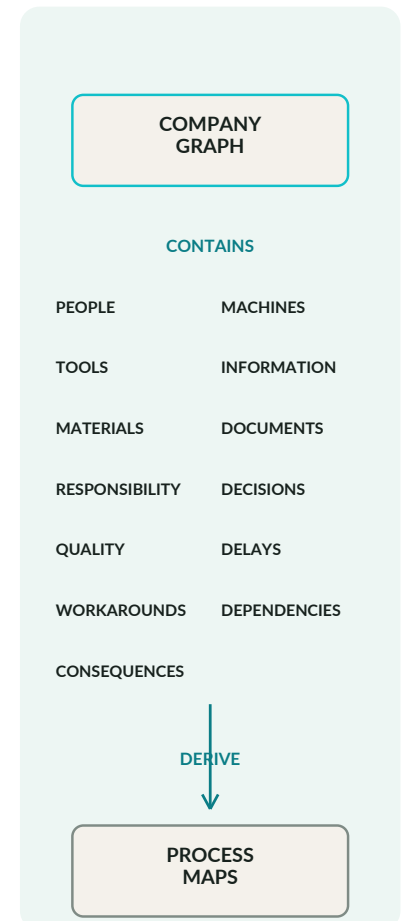
Most consulting work begins with predefined processes. That is already a mistake.

A process map is only a selected view of reality. It excludes everything its author did not know, did not notice or did not want to include. I do not reconstruct the company from process maps. I reconstruct the company first and derive its processes afterward.

The graph contains people, machines, tools, information, materials, documents, responsibility, decisions, quality, delays, unofficial workarounds, dependencies and consequences. Only when these elements exist within one model can the company be understood as a system.

A local inefficiency may be caused by a decision made three departments away. A quality problem may begin in procurement. A production delay may be created by information architecture. An overloaded employee may be compensating for a structural defect invisible in every management report.

Optimizing one process without seeing the full graph can improve one metric while damaging the company. That is not optimization. It is hiding the cost somewhere else.



Management Comes Last

I speak with management only after I have mapped the organization. At that point, I am no longer asking executives how the company works. I already know. I compare their mental model with the operating system I have reconstructed.

The gap is often uncomfortable. It explains why companies continue treating structural failures as isolated incidents and why the same problems return under different names. Management sees departments; I see dependencies. Management sees reports; I see the conditions that produced them. Management sees symptoms; I see the architecture manufacturing them.

DEPARTMENTS

DEPENDENCIES

REPORTS

CONDITIONS

SYMPTOMS

ARCHITECTURE

*Management sees departments; I see dependencies.
Management sees reports; I see the conditions that produced them.
Management sees symptoms; I see the architecture manufacturing them.*

works in three ways. I can map the existing architecture. I can map it and design a new one. Or I can map it, redesign it and take responsibility for full implementation.

These are not modular consulting packages from which a company selects a few convenient pieces. I do not analyze one department while pretending the surrounding system does not matter. I do not produce a reduced version designed to fit a smaller budget, and I do not negotiate the scope downward until the work becomes harmless, comfortable and useless. The client accepts the conditions or does not. That is the decision.

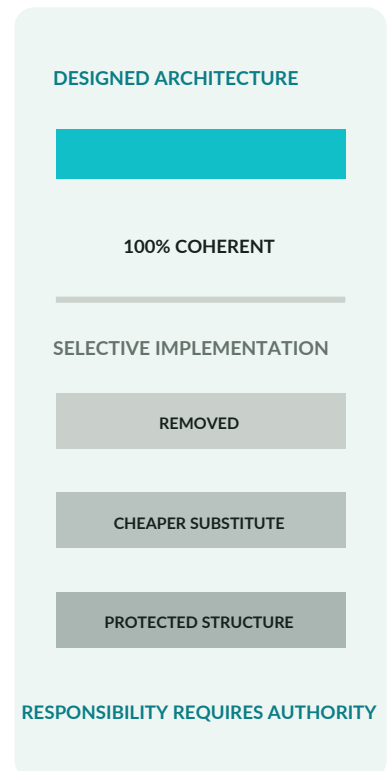
Architecture Cannot Be Implemented in Fragments

Designing an architecture and implementing it are two different responsibilities. In many cases, the safer engagement ends with the completed design.

Sometimes a company implements only part of what was designed. It removes difficult elements, replaces essential components with cheaper substitutes and protects the internal structures that created the original problem. Then it says the architecture failed.

It did not. The company refused to implement it.

What failed was a damaged imitation carrying the name of the original design. You cannot implement one-third of an architecture and blame the architect for the result. That is why I accept responsibility for implementation only when I also receive the authority required to protect the coherence of the system. Responsibility without authority is theater, and I do not participate in theater.



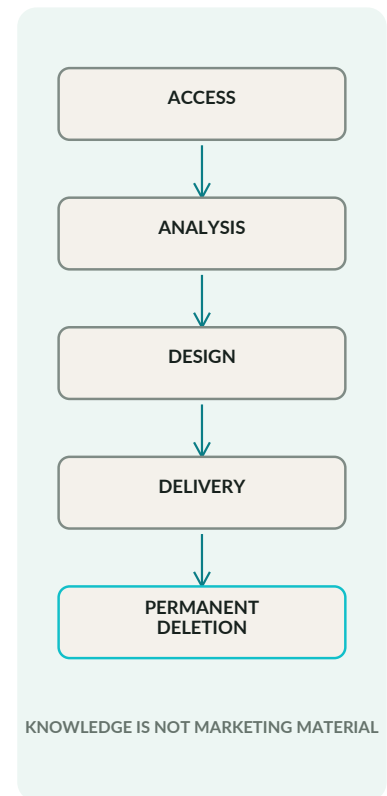
Confidentiality Begins With Silence

This work gives me access to the deepest operational layer of a company. I may see more of its real operating structure in one place than most of its managers have ever seen together. That knowledge is not marketing material.

does not disclose who it works for, who it has previously worked for or which companies have been optimized. I do not publish conventional case studies. A client's weaknesses, internal structure, redesigned architecture and competitive advantage belong to the client.

Imagine that I enter a pharmaceutical factory owned by a globally known company and we announce it publicly. The market immediately knows that something important is happening inside that organization. Competitors do not need the final design. They only need to know where to begin asking questions, whom to approach and which part of the company may contain valuable information.

Industrial espionage is not a theoretical risk. It is older than modern consulting. The strongest first layer of protection is that nobody knows an intervention is taking place.



Confidentiality Begins With Silence - continued

Everyone working within [this firm](#) is bound by severe confidentiality obligations, and client information is physically protected throughout the engagement. When the work is complete and the agreed materials have been delivered, the data is permanently deleted from my equipment. I do not retain a second copy, a private archive or material for future promotion.

The client must secure what it receives. If those materials are later lost, I cannot send them again because there is nothing left to send. Confidentiality is not a clause added to the contract. It is part of the architecture of the engagement.

NO CLIENT DISCLOSURE

NO CONVENTIONAL CASE STUDY

NO PRIVATE ARCHIVE

NO SECOND COPY

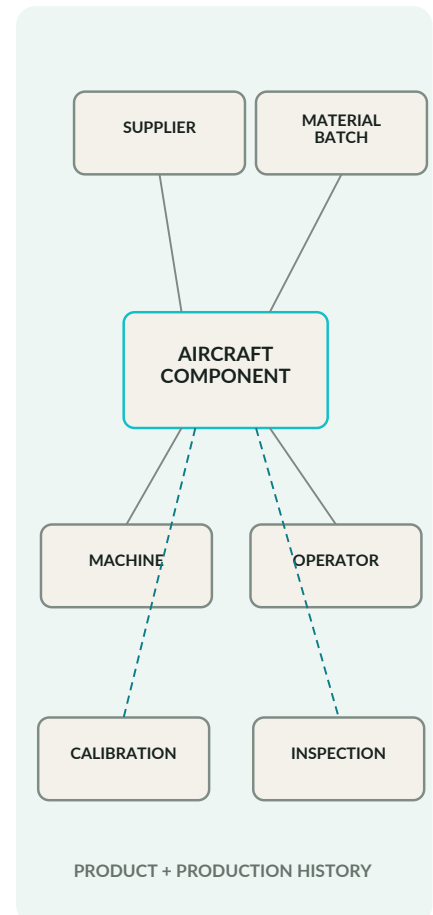
Why Aircraft Production Is an Architectural Problem

Aircraft production is a useful public example because it demonstrates how quickly conventional process thinking becomes inadequate.

Consider a defect discovered several years after an aircraft enters service. A conventional investigation may begin with the failed component, the production batch and the inspection record. An architectural investigation must go further. It must be possible to connect the component with the material supplier, manufacturing batch, workstation, machine, tool, operator, calibration history, environmental conditions, inspection sequence, software version, maintenance record and every other aircraft exposed to the same combination of conditions.

The problem is not merely whether the failed part passed quality control. The problem is whether the production system preserved enough connected information to reconstruct the conditions under which it was created.

Suppose several defects appear in components that were manufactured on different dates and installed in different aircraft. They may initially look unrelated. Yet the graph may show that all of them passed through the same workstation after a particular maintenance intervention, were produced using material from the same supplier batch, or were inspected with equipment operating under the same calibration error.



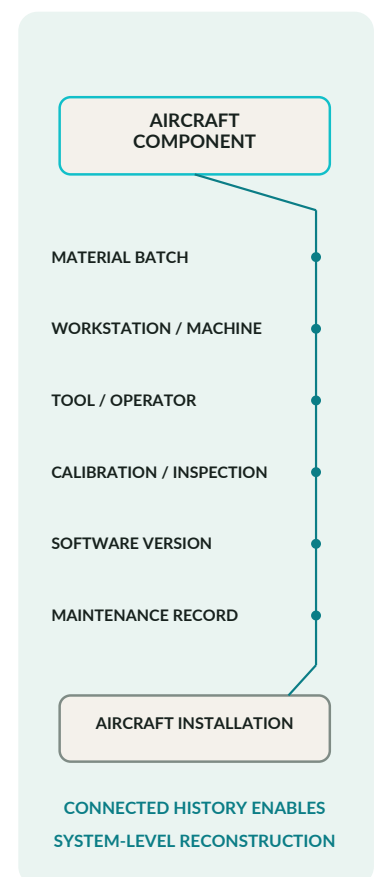
Why Aircraft Production Is an Architectural Problem - continued

Without an integrated architecture, those relationships remain scattered across departments, databases, paper records and individual memories. Production sees one fragment. Quality sees another. Procurement sees another. Maintenance sees another. Each department may perform its own investigation correctly and still fail to see the system that connects the evidence.

A properly designed architecture makes the product inseparable from its production history. The aircraft component is no longer just an object with a serial number. It becomes a node connected to the entire set of conditions that created, inspected, moved and approved it.

That principle is not limited to aviation. It applies wherever a failure can have serious human, financial or strategic consequences. I can explain the architectural logic publicly. What I will not disclose is whether I applied it for a particular company, where it was implemented or what competitive result it created.

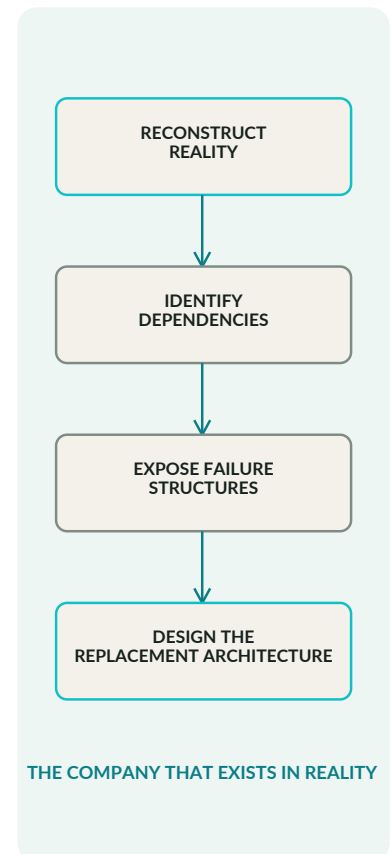
Capability can be demonstrated without turning a client into evidence.



What I Actually Do

I do not enter a company to produce another polite list of recommendations. I reconstruct its real operating architecture. I show how the organization actually creates value, where it loses control, which dependencies remain invisible and which structures continue generating the same failures.

Then, when required, I design the system that should replace them.



Most management boards manage the company that exists in reports. My work begins with the company that exists in reality.

**MOST MANAGEMENT BOARDS
MANAGE THE COMPANY THAT
EXISTS IN REPORTS.**

**MY WORK BEGINS WITH THE
COMPANY THAT EXISTS IN REALITY.**

The company is the graph.

MATEUSZ CHRZANOWSKI
SYSTEM OF SYSTEMS ARCHITECT



Chrzanowski