
ORGANIZATION FIRST. TECHNOLOGY SECOND.

Modeling Processes in the Age of AI.

The CLEAR Framework

Five roles, two rules and one foundation for processes in which people and AI agents work side by side.

C

CONSULTS

L

LEADS

E

EXECUTES

A

APPROVES

R

REVIEWS

PUBLICATION

06 October 2026

SERIES

Organization Design for
AI

AUTHOR

Maximilian Stein

PUBLISHER

HandsOn — AI Strategy
& Governance
Consulting

ABSTRACT

Every actor in the process is a person. That assumption no longer holds.

Every well-run company models its processes. Flowcharts & Swimlanes show the steps and handoffs and a responsibility table (usually RACI) shows who does what. These models are how you standardize work, onboard people and find what's broken.

All of them share one assumption: every actor in the process is a person.

That assumption no longer holds. AI systems now draft the email, update the CRM and file the task. A process model that leaves them

out describes a company that no longer exists. And a model that simply adds them as “a tool” hides the most important question of all: who is accountable when the machine does the work?

The CLEAR Framework is a way to model processes for this new reality. This article explains what changes when AI becomes part of a process, how CLEAR models it, how it shows where the time in a process really goes, what that looks like in a real case, and what you get out of it.

EXECUTIVE SUMMARY · FIVE FINDINGS

01 Classic process models hide who is accountable when AI does the work.

AI steps either disappear inside a human's box or get a “System” box with no one attached to it.

02 CLEAR replaces RACI's four letters with five roles and two rules.

Consults, Leads, Executes, Approves, Reviews. Leadership is human-only, and no AI chain runs without a checkpoint.

03 The Context Layer decides how much you can actually hand over.

Information becomes context when a machine can find it, trust it, and is allowed to use it.

04 AI shortens working time; human gates still set the lead time.

Every Approve is a point where work stops until a human gets to it. That is where the bottlenecks sit.

05 The process model becomes the agent's configuration.

The agentic steps within CLEAR can be used to configure the agent.

CONTENTS

What this paper covers.

FRONT MATTER

00	Abstract & Executive Summary	02
----	------------------------------	----

PART I · THE PREMISE

01	What Changes When AI Joins the Process	05
	Three questions for every step	

PART II · THE FRAMEWORK

02	Five Roles, Two Rules, One Foundation	07
	Consults · Leads · Executes · Approves · Reviews	
03	The Foundation: the Context Layer	08
	Source of truth · labels · permissions · connections · write-back	
04	The Time Dimension	09
	Lead times today and in the target state · human bottlenecks	

PART III · IN PRACTICE

05	How Modeling with CLEAR Works	12
	Five passes	
06	A Live Example: Modeling an AI Chief of Staff	13
	The CLEAR matrix · six design decisions · lead times	

CLOSING

07	What a CLEAR Process Model Gives You	16
	Eight outcomes	
08	Where to Start	18
	Five questions for your first process	



Who is accountable when the machine does the work?

A process model that leaves AI out describes a company that no longer exists.

IN THIS PART

01 What Changes When AI Joins the Process

05

SECTION 01 · THE SHIFT

What Changes When AI Joins the Process.

Why classic process models lose track of accountability once a machine does part of the work.

In a classic process model, an AI step usually ends up in one of two places. Either it disappears inside a human's box ("Sales prepares the proposal", even though an assistant

writes most of it), or it gets a box of its own labeled "System" with no one attached to it.

Both hide what now matters most. A process model that includes AI has to answer three questions for every step.

EXHIBIT 1

Three questions every step of an AI-supported process has to answer

1

Who leads?

Which person owns the outcome, however much of the work a machine does?

2

Who or what executes?

Is this step done by a person, an AI agent, or both together?

3

Where must a human sign off?

At which points does work stop until someone checks it?

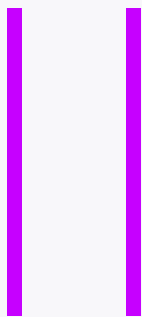
Source: HandsOn CLEAR Framework v1.2

Classic models were never built to answer these. Companies bringing in AI one tool at a time feel the result. Either nobody dares to let the AI do anything meaningful, because nobody knows who would answer for a mistake. Or the AI quietly does more and more, and accountability dissolves along the way.

TWO FAILURE MODES

Paralysis. Nobody lets the AI do anything meaningful.

Drift. The AI does more and more; accountability dissolves.



Five letters, two rules, one table per process.

That is the model. But a model is only as good as the ground it stands on.

IN THIS PART

02	Five Roles, Two Rules, One Foundation	07
03	The Foundation: the Context Layer	08
04	The Time Dimension	09

SECTION 02 · THE FRAMEWORK

Five Roles, Two Rules, One Foundation.

CLEAR keeps what made process models useful: a plain table anyone can fill in, one row per process step.

What changes is the columns. Instead of the four RACI letters, every step is modeled with five roles:

EXHIBIT 2

The five CLEAR roles

CODE	ROLE	WHAT IT MEANS	HUMAN OR AI?
C	Consults	Gives input to the work: a colleague, an expert or an agent. Data sits in the Context Layer.	Either
L	Leads	Owens the outcome. Exactly one per step.	Always a human
E	Executes	Does the work.	Either
A	Approves	The gate: nothing moves on without this sign-off.	Human wherever a decision matters
R	Reviews	Checks finished work afterwards, often as a sample.	Either, as designed

Source: HandsOn CLEAR Framework v1.2

Two rules apply to every model, with no exceptions:

RULE 1**Leadership is human-only.**

An AI system can execute, consult or pre-check. It never leads. Someone with a name is always accountable for the outcome.

RULE 2**No AI chain runs without a checkpoint.**

When one agent hands work to another, there is either a human touchpoint or a defined, logged machine check between them.

Five letters, two rules, one table per process. That is the model. But a model is only as good as the ground it stands on, which is why CLEAR has a third building block: the Context Layer, covered below.

SECTION 03 · THE FOUNDATION

The Foundation: the Context Layer.

Roles tell you who does what. They don't tell you whether the work *can* be done well.

A person can ask a colleague, notice that a spreadsheet is outdated or remember what the client said last week. An AI agent can only work with what it can reach through your systems.

That is why every CLEAR model sits on a **Context Layer**: the system layer underneath the process. It is built on one simple idea:

“ Information becomes context when a machine can find it, trust it, and is allowed to use it.

In practice, that means five things:

EXHIBIT 3

The five components of the Context Layer

COMPONENT	THE QUESTION IT ANSWERS
01 Source of truth	For every piece of information, which <i>one</i> system holds the valid version? Agents read from there, never from copies.
02 Labels	Does every document and record show who owns it, how current it is and how confidential it is?
03 Permissions	Can every agent access exactly what its tasks need, and nothing more, traceable back to a responsible person?
04 Connections	Do agents reach your systems through proper, controlled interfaces, rather than exports, uploads and copy-paste?
05 Write-back	Do the results of the work flow back into the systems in a form the next step, human or AI, can use? If not, the layer decays.

Source: HandsOn CLEAR Framework v1.2

The Context Layer decides how much you can actually hand over. If an agent is meant to work on its own, the information it depends on has to meet all five points. If it doesn't, a human checks every output until the gap is closed. This gives you one of the most useful sentences in any AI project:

“Your process model says the agent can do this alone. Your systems say a human still has to check every result.”

The gap between the two is your roadmap, and it usually lies in the data and systems, not in the AI.

SECTION 04 · LEAD TIME

The Time Dimension: Lead Times Today and in the Target State.

A process model that only shows *who* does what still misses the question every managing director asks first: *how long does it take?*

So in CLEAR, every step also carries time. The principle is simple. Every step has two kinds of time:

Working time: someone, human or AI, is actually doing the work.

Waiting time: the work sits in an inbox, a queue or a to-do list until someone gets to it.

Add both up across all steps and you get the **lead time** of the process: how long it takes

from trigger to result. In most processes, waiting time is by far the larger part. The work itself takes minutes; the waiting takes days.

CLEAR compares this lead time in two states: **today**, the process as it runs now, with its real working and waiting times; and the **target state**, the redesigned process, with AI executing where the model allows it and human checkpoints where they are needed.

EXHIBIT 4

AI shrinks working time; waiting at human gates remains, and an unready Context Layer adds checks back in

■ Working time ■ Waiting time ■ Added human checks

Today

As it runs now

**Target state as planned**

AI executes where allowed

**Target state as permitted**

What the Context Layer allows



Schematic, not to scale. Source: HandsOn CLEAR Framework v1.2; logic mirrors the CLEAR Process Simulator

That comparison shows something AI enthusiasm usually hides. AI shortens *working time* dramatically: a draft that took an hour takes a minute. But it barely touches *waiting time* at the steps that still need a person.

Every **Approve** is a point where work stops until a human gets to it. If that person is in meetings all day, an agent that finishes in seconds still waits for hours.

SECTION 04 · CONTINUED

Where Work Waits for a Person.

So the lead-time view tells you exactly where the human bottlenecks are.

BOTTLENECK 01**Approval queues**

Too many outputs need sign-off from too few people.

BOTTLENECK 02**Concentration**

All approvals land on one desk, typically the boss.

BOTTLENECK 03**Handoffs and batching**

Work waits for the weekly meeting, the next shift or the end of the month.

BOTTLENECK 04**Office hours**

The agent works at night, the approver doesn't.

PLANNED VERSUS PERMITTED

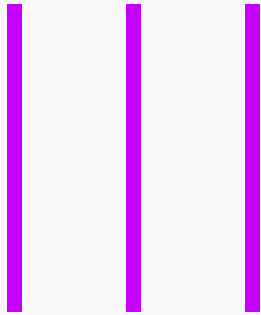
There is one more honest step. The target state you *plan* and the target state you are *allowed* to run are not always the same. If the Context Layer isn't ready, a human has to check every output, which adds waiting time back in. So CLEAR always calculates the target state as permitted, not just as drawn.

THE HANDSON POSITION

The fix for a human bottleneck is rarely “remove the human”. It is better design: a review deadline so gates don't turn into queues, a Review afterwards instead of an Approve before where the risk allows it, approvals bundled and brought to the person where they already are, and no more than one checkpoint per phase.

“ If that person is in meetings all day, an agent that finishes in seconds still waits for hours.

— THE LEAD-TIME VIEW



The process model became the agent's configuration.

We didn't write a separate specification. We read it straight off the model.

IN THIS PART

05	How Modeling with CLEAR Works	12
06	A Live Example: Modeling an AI Chief of Staff	13

SECTION 05 · METHOD

How Modeling with CLEAR Works.

Modeling a process with CLEAR takes five passes.

EXHIBIT 5

Five passes from process steps to business case

1	List the steps from the outcome, not from today's routine.	Many current steps only exist because people work in batches, hand over by email or keep office hours. An AI-supported process doesn't have to inherit them.
2	Assign the five roles to every step.	One Lead, always human. An Executor, human or AI. Consult, Approve and Review where they are needed.
3	Place the checkpoints on purpose.	Put the Approve gate where a mistake is expensive, irreversible or visible to a client. Everywhere else, a lighter Review is enough.
4	Check the context underneath.	For every step an AI executes, ask where its information comes from and whether it meets the five Context Layer points. Where it doesn't, keep a human gate in place and put the gap on the roadmap.
5	Put time on every step.	Estimate working and waiting time today and in the target state. Compare the lead times and mark where work waits for a person. Those are your bottlenecks, and your business case.

Source: HandsOn CLEAR Framework v1.2

THE RESULT

The result is a model that shows the process as it really runs, including exactly where the line between human and machine sits, what the systems underneath can support today, and where the time actually goes.

SECTION 06 · CASE EXAMPLE

A Live Example: Modeling an AI Chief of Staff.

Here is what that looks like in practice.

The client was an owner-led agency: campaigns, content production and talent management running in parallel, with one managing director keeping it all together. The MD's day was split across four systems: email and calendar, a CRM for the sales pipeline, a project tool for the team's production tasks, and a wiki of briefs and playbooks.

The goal was an AI chief of staff: an agent the MD talks to over a messenger, which reads across all four systems, briefs the day, keeps track of deadlines and handles the admin.

The tempting approach is to connect the agent to everything and "let it help". We did the opposite: we mapped the systems and modeled the process first.

EXHIBIT 6

The CLEAR model of the AI chief of staff

STEP	CLEAR ROLES · WHO					CONTEXT LAYER · WHICH SYSTEMS	
	C · CONSULTS	L	E · EXECUTES	A · APPROVES	R · REVIEWS	READS FROM	WRITES BACK TO
Morning brief: what's about to slip	–	MD	Agent	–	MD, while reading it	Calendar, CRM, project tool, wiki	–
Capture a personal reminder	–	MD	Agent	–	MD, via the agent's confirmation	–	Calendar
Create a production task for the team	Project lead (capacity)	MD	Agent	–	MD, via the agent's confirmation	Brief in the wiki	Project tool
Log a meeting summary in the CRM	–	MD	Agent	–	MD, via the agent's confirmation	Meeting notes	CRM
Draft an email to a client	Project lead (status)	MD	Agent drafts	MD sends	–	CRM history, past threads	Email outbox
Create a deal or move it in the pipeline	–	MD	Agent prepares	MD confirms	–	CRM	CRM
Update briefs and playbooks	Agent (flags outdated content)	MD	Team (humans only)	–	–	Project tool, CRM	Wiki

Note: Fictionalized from a small-organization engagement; client identity removed. Shaded rows carry an Approve gate. C = who gives input; Context Layer = systems read and written.

SECTION 06 · CONTINUED

Six Design Decisions.

The model made six design decisions visible.

DECISION 1

The MD leads everything, and that is the point.

In a small company, one person often holds L, A and some of the E. At first that looks like the model adds nothing. It doesn't, because the value in a small company isn't dividing work among people. It is drawing the **line between human and machine** clearly and putting the checkpoints in the right places.

DECISION 2

Reversible work runs without a gate.

Reminders, internal tasks and meeting notes all have an empty A column. If the agent gets one wrong, it takes ten seconds to fix. So the agent just does it, then reports exactly what it wrote and where. That confirmation is the Review.

DECISION 3

Anything that leaves the building or touches money has an Approve.

The agent never sends an email. It drafts, and the MD sends. It never moves a deal by itself. It prepares the change, and the MD confirms with a short "ok". The line sits where a mistake becomes expensive or visible to a client.

DECISION 4

Some systems are read-only.

The agent reads from the wiki, but only the team writes back to it. The agent may flag outdated content as a Consult, never change it. That one cell became a technical setting: read-only access.

DECISION 5

The Context Layer came first.

Before assigning a single role, we set the source of truth for each kind of information: appointments live in the calendar, deals in the CRM, team tasks in the project tool, briefs in the wiki. The agent reads live from those systems and never answers from memory or copies. It gets its own scoped access to each system. And every action it takes writes back: meeting summaries land in the CRM, tasks in the project tool, so the next step starts from current information. Without that groundwork, the roles in Exhibit 6 would have been wishful thinking.

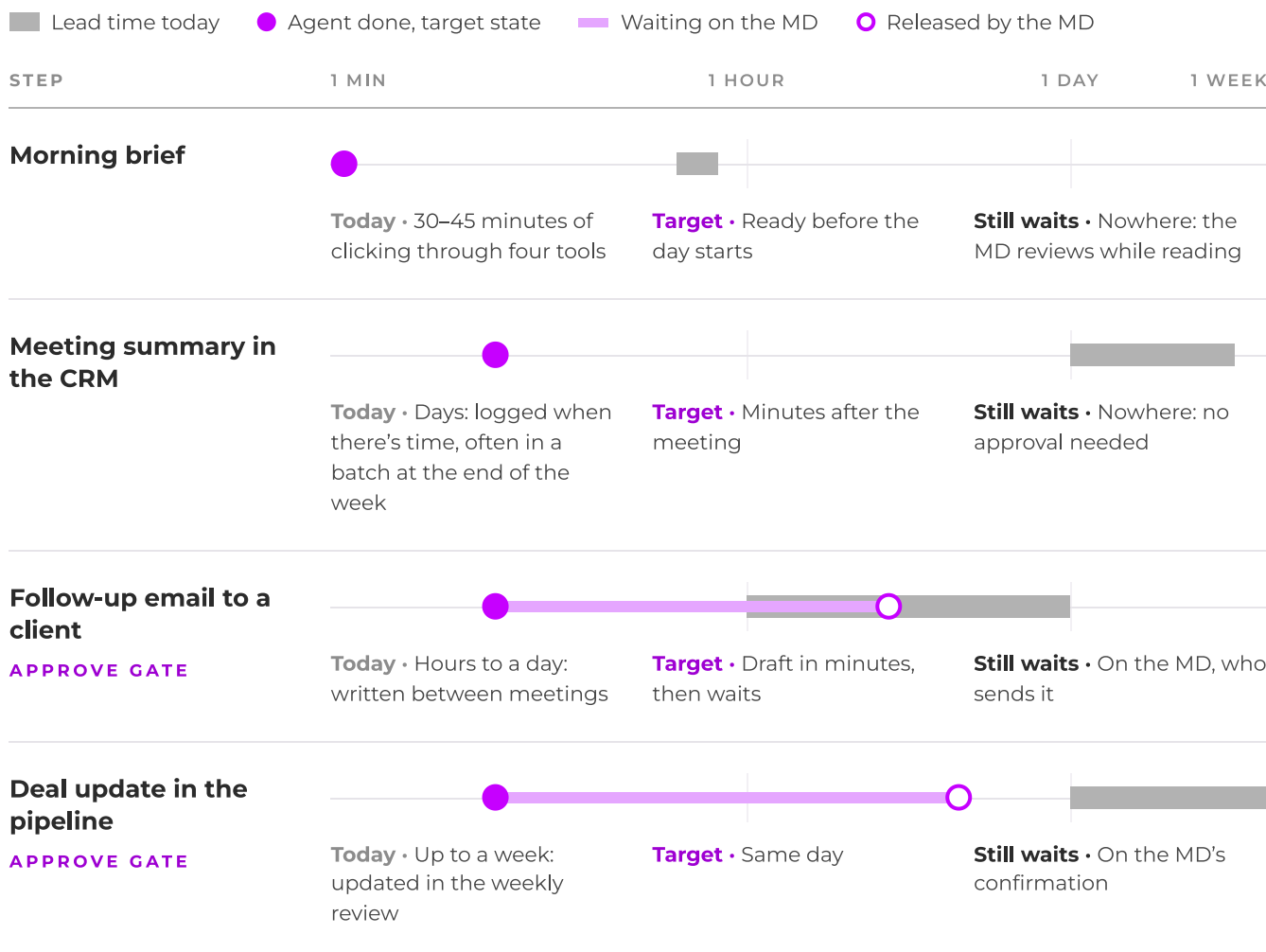
DECISION 6

The bottleneck was the MD, not the machine.

Putting time on the steps showed the pattern clearly. Exhibit 7 overleaf shows typical orders of magnitude for a setup like this (illustrative, not measured).

EXHIBIT 7

Where there is no Approve, lead time collapses from days to minutes



Note: Typical orders of magnitude for a setup like this; illustrative, not measured. Logarithmic time scale.

Where there is no Approve, lead time collapses from days to minutes. Where there is one, lead time now depends almost entirely on how fast the MD responds. Every approval in the model lands on the same desk. So we designed for that: approvals come to the MD in the messenger they already use, and a short “ok” is enough. The gate stays, but the queue in front of it shrinks.

The result: **the process model became the agent's configuration.** Every E cell defined what the agent is allowed to do. Every A cell became a confirmation step built into the tool. Every C cell named whose input the agent asks for. The Context Layer columns set which systems it reads from and writes back to. We didn't write a separate specification. We read it straight off the model.

SECTION 07 · THE VALUE

What a CLEAR Process Model Gives You.

Eight outcomes, from one standard across every process to a model that lives in your systems.

-
- | | | |
|-----------|---|--|
| 01 | One standard across every process | Sales, finance, operations and HR are all modeled with the same five letters and the same two rules. Processes become comparable, so you can see at a glance where AI already works, where humans sign off and where the gaps are. New team members, auditors and partners read one format instead of five. |
| <hr/> | | |
| 02 | Agents built from the model, not the other way round | Most AI projects start with a tool and look for a use. CLEAR reverses that. Once a process is modeled, the agent's job description is already written: the E cells are its tasks, the A cells its limits, the Context Layer its data access. That makes agents faster to build, easier to explain and much easier to change. |
| <hr/> | | |
| 03 | Accountability that never dissolves | Rule 1 means every step has one named person who owns the outcome, however much of the work a machine does. That is good management, and it is the core of what the EU AI Act means by human oversight. CLEAR builds it into every process step. |
| <hr/> | | |
| 04 | Oversight where it matters, not everywhere | A sign-off on every single output looks safe, but within weeks people approve without really looking. A CLEAR model places the Approve gate where a mistake would be costly and uses lighter Reviews everywhere else. Fewer checkpoints, taken seriously, beat many that nobody reads. |
-

SECTION 07 · CONTINUED

-
- 05 A model that grows with trust**
- As an agent proves itself, a step can move from “MD approves” to “MD reviews afterwards”, without remodeling the process. The table stays the same. Only the checkpoint changes. Your AI roadmap becomes a series of small, deliberate steps instead of one big leap.
-
- 06 A business case you can calculate**
- Because every step carries working and waiting time, the effect of a change can be specifically measured: lead time today versus lead time in the target state, as permitted by the Context Layer. And you see where the time really goes, which is usually not the work itself but people waiting on people.
-
- 07 Honesty about what’s actually ready**
- The Context Layer shows you why an AI project stalls before you spend money on it. Instead of blaming the model or the vendor, you see the real bottleneck, which is usually scattered data, unclear ownership or missing access rules. That turns a vague “AI isn’t working for us” into a concrete, prioritized to-do list.
-
- 08 A model that lives in your systems**
- Most process models go out of date the day after the workshop. A CLEAR model counts as done only once it is built into the systems where work happens: CRM fields, workflow rules, agent configurations. A model that *is* the configuration can’t drift away from reality without anyone noticing.
-

“ Fewer checkpoints, taken seriously, beat many that nobody reads.

— THE CLEAR FRAMEWORK

CLOSING

Where to Start.

Pick one process where AI already does some of the work, or where you want it to. Write down the steps.

FOR EACH ONE, ASK THE THREE QUESTIONS

- 01 Who leads?
- 02 Who or what executes?
- 03 Where must a human sign off?

THEN ASK A FOURTH, AND A FIFTH

- 04 Where does the information for this step come from, and can a machine find it, trust it and use it?
- 05 How long does this step take today, and how much of that is just waiting?

You'll see quickly where accountability is clear, where it has quietly gone missing, and where your process is really waiting for a person. That model is the starting point for every agent you build afterwards.

ABOUT HANDSON

CLEAR is part of the HandsOn AI Operating Model. If you'd like to model your first process together, let's talk.

wearehandson.de

