

CONCEPTS · THE OPERATING LAYER

The Operating Layer.

One worked case — Management of Change at a high-hazard refinery — and the substrate that makes the work queryable rather than assembled.

A TELLEFSEN BRIEF · 10-PAGE EDITORIAL

CONTENTS

How to read this brief.

01	The pattern When the work spans systems, requires lineage, and carries regulatory teeth.	p. 2
02	Today — eight to twelve days of hunting What it looks like to scope a single Management of Change by hand.	p. 3
03	On the Operating Layer The same request, composed in under thirty seconds.	p. 4
04	Two ways to do the same work A side-by-side of what changes.	p. 6
05	Why this is architecture, not features Three structural differences from a chatbot, copilot, or platform.	p. 7
06	The four components Context Graph, Orchestration, Governance, Connectors.	p. 8
07	Compounding Why the substrate matters more than any single use case.	p. 9
08	How to engage Tellefsen Sprint, Build, Retain.	p. 10

AI does not solve this. A chatbot over your documents does not solve this. A copilot inside one of your systems does not solve this. None of them solve it because none of them are the missing layer — the substrate that connects the systems, holds the lineage, and lets the work be queried rather than assembled. That layer is what Tellefsen builds. We call it the Operating Layer.

01 · THE PATTERN

Where this kind of work lives.

Every operationally complex organisation has work like this. A change in one place must be traced through dozens of others. The trace touches multiple systems, draws on years of history, must be auditable to a regulator, and must compound learning across the organisation so the next person doing similar work is not starting from zero. Today, this kind of work is done by hunting — by people moving between systems, assembling context by hand, and remembering what they can about precedent.

Banks call it model risk management. Ministries call it policy impact assessment. Utilities call it grid reliability review. Aviation operators call it safety management. Healthcare systems call it clinical change control. The names differ; the structure is identical. In each case the work is regulated, multi-system, lineage-dependent, and currently done by assembly rather than by query.

Rather than describe the Operating Layer abstractly, we will show it through one piece of work — Management of Change at a high-hazard refinery. The pattern is general; this case is concrete. If you do not work in process industries, mentally substitute the equivalent in your own world. The architecture is the same.

A major Gulf integrated energy operator. A refinery in Sohar. A reliability engineer who needs to raise the operating pressure of a heat exchanger.

02

Today — eight to twelve days of hunting.

A reliability engineer at a Gulf refinery wants to raise the operating pressure of heat exchanger E-12503 in the Hydrocracker unit from 22 barg to 26 barg, to recover throughput lost since the last turnaround. This is a process change. Process Safety Management regulations require a formal Management of Change — an MOC.

This is what the next two weeks of her life look like.

- 01 She fills out the MOC form in the HSE system.
- 02 She searches SharePoint for the latest P&IDs. Three revisions exist. She loses a day confirming which is current.
- 03 She opens the HAZOP study from 2018. She has to read it to determine whether the requested pressure is inside the study envelope. It isn't.
- 04 She searches SAP PM for past work orders on E-12503. Fourteen entries since 2014. She reads each one.
- 05 She asks the senior process engineer — retiring next year — whether anything similar has been done at any of the operator's other facilities. He half-remembers a change at the methanol plant. He isn't sure.
- 06 She searches the procedure library for SOPs and emergency response procedures that reference the unit. She finds three. They will need updating.
- 07 She identifies which operators are competent on the unit and whether retraining is required.
- 08 She waits for the weekly multi-disciplinary review meeting.
- 09 She assembles the packet for the Pre-Startup Safety Review.

Elapsed time, MOC submission to scope-complete review packet: eight to twelve working days. Most of it is hunting, cross-referencing, and waiting. None of it requires judgement that a person should be doing by hand.

The data exists. The substrate to connect it does not.

03

The same request, on the Operating Layer.

The engineer enters the request. Within thirty seconds, the substrate composes the answer.

MOC REQUEST

Raise E-12503 operating pressure from 22 to 26 barg.

~28 sec · 8 sources

AFFECTED P&IDS

GRAPH

Three drawings identified, current revisions confirmed. Four downstream equipment items whose design basis is affected by the proposed pressure change.

PROVENANCE · Document graph · Engineering DMS · revision-aware

EQUIPMENT HISTORY · E-12503

GRAPH

Fourteen prior work orders since 2014. Two prior MOCs: 2019 tube bundle replacement, 2022 control valve upgrade. Canonical-plus-version timeline available.

PROVENANCE · SAP PM · canonical equipment record · 12 years

CROSS-ENTITY PRECEDENT

GRAPH

Similar change executed at the operator's methanol facility in 2021. Full MOC record, scope, lessons learned. Two findings worth reviewing.

PROVENANCE · MOC graph · cross-facility · authority: HSE

HAZOP ENVELOPE CHECK

SCRIPT

HAZ-2018-HCU-04 covers operating pressure up to 24 barg only. Requested 26 barg is outside study envelope. HAZOP re-evaluation required before approval.

PROVENANCE · HAZOP database · deterministic envelope compare

COMPOSED RESPONSE · CONTINUED

Four more sources, composed in the same pass.

<p>DOCUMENTS REQUIRING UPDATE GRAPH</p> <hr/> <p>SOP-OQBI-0142 (Hydrocracker operations), alarm setpoint table A-HCU-007, emergency response procedure ERP-HCU-03. All linked, all flagged.</p> <p>PROVENANCE · Procedure graph · equipment-linked references</p>	<p>TRAINING IMPACT GRAPH</p> <hr/> <p>Eleven operators competent on the Hydrocracker need procedural retraining. Roles and current training status visible. Assignment is by the unit supervisor, not by the system.</p> <p>PROVENANCE · Competency system · role-linked · current as of today</p>
<p>APPROVAL CHAIN GRAPH</p> <hr/> <p>Multi-disciplinary reviewers required for a pressure change of this magnitude, per the operator's MOC procedure. Current MOC backlog visible per reviewer.</p> <p>PROVENANCE · MOC procedure · live reviewer workload</p>	<p>DRAFT MOC SCOPE DOCUMENT AI</p> <hr/> <p>Auto-generated from the graph above. Every claim cites a source node and the authority that established it. Ready for the engineer to review and edit.</p> <p>PROVENANCE · Composed by AI · all citations link to graph nodes above</p>

Eight-to-twelve days, compressed into one morning. With better completeness than the hand-assembled version — because nothing was missed in the hunting.

04

Two ways to do the same work.

TODAY

8-12

WORKING DAYS · MOC SCOPING

- Hunting across systems
- Cross-referencing by hand
- Remembering precedent
- Manual document discovery
- Waiting for review meetings
- Assembling the packet

Most of the time is not judgement.

ON THE OPERATING LAYER

<30

SECONDS · TO ASSEMBLED ANSWER

- Graph traversal across systems
- Cross-entity precedent surfacing
- Canonical-plus-version on equipment state
- Deterministic safety envelope checks
- AI for what genuinely needs it
- Provenance on every claim

The judgement is still the engineer's.

05

Why this is architecture, not features.

Every AI product can show a demo. The question is what is underneath. Three structural differences distinguish what was just shown from a chatbot, a copilot, or a private-deployment AI platform.

01. The data is connected, not searched.

A chatbot over your documents still searches. A copilot inside your HSE system still searches. What was shown above does not search — it traverses. The P&IDs, the equipment history, the procedures, the training records, the work orders are all nodes in one graph, connected by their real operational relationships. The substrate makes the connection structural. The reader is not waiting for a search result; the answer is composed from the graph in a single pass.

02. The execution mode is chosen, not assumed.

Most AI products are AI-first by default. Every question is a model call. That is wrong for safety-critical work. The HAZOP envelope check is not a judgement; it is a logical comparison and it must never be approximated. A script does it deterministically. The cross-entity precedent is a graph traversal — no AI needed. The draft scope document is genuinely a language task — AI is the right tool. The substrate decides per sub-question.

03. The lineage is structural, not assembled.

Every claim shown to the engineer carries provenance — the source node, the authority, the timestamp. When she submits the MOC packet for review, the reviewer is not asked to trust the system; they can trace any element back to its source. When the regulator audits a year later, the lineage is queryable, not reconstructed.

These are architectural properties of the layer underneath. They cannot be retrofitted onto a chatbot. They cannot be added to a deployment platform. They are what an Operating Layer is for.

06

The architecture, briefly.

The Operating Layer is what Tellefsen builds. Every Operating Layer has the same four components. In this case, here is what each one is.

CONTEXT GRAPH

The ontology layer

The model of the business — entities, relationships, lifecycle events. The ontology that turns raw system records into something that can be reasoned over.

IN THIS CASE

Facilities, units, equipment, MOCs, procedures, HAZOPs, training records, work orders — all connected nodes.

ORCHESTRATION

The runtime

How agents, scripts and AI models are composed, sequenced, and handed off — with human-in-the-loop gates where it matters.

IN THIS CASE

The runtime that decides per sub-question whether the right tool is a graph traversal, a script, or an AI call.

GOVERNANCE

The control layer

Permissions, audit, redaction, approval gates, evaluation hooks, structural lineage on every derived claim.

IN THIS CASE

Recommend-and-human-approves on every action. Provenance carried on every block of the response.

CONNECTORS

The bridges

Read-from-and-write-back-to bridges to the systems of record — never replacing them, always reading their truth.

IN THIS CASE

SAP PM, the HSE system, the document estate, the HAZOP software, the training system — all bridged, none replaced.

Same four components in every Tellefsen engagement. Different ontology, different substrate, different sector. Same architecture.

07

Compounding.

There is a second story buried in this case. The MOC at the refinery becomes a node in the graph. A year later, the same operator's gas networks business is scoping a similar change at a compression station. The substrate surfaces the refinery's experience — the scope decisions, the lessons learned, the things that went wrong — as the top precedent, with full provenance. The work compounds. Not because anyone wrote a knowledge base. Because the architecture connects the records. This is the difference between an organisation that runs on systems and an organisation that runs on an Operating Layer.

The substrate is the foundation. The work that runs on it compounds for as long as it exists.

ENGAGEMENT MODEL

Three stages. Fixed price. Operating Layer Map signed off in 8–12 weeks.

SPRINT

8–12 weeks · fixed price

We map your Operating Layer: the ontology, the systems of record, the connectors, the first use case. You receive a signed-off Operating Layer Map.

BUILD

Quarter-by-quarter

We build the substrate — Context Graph, Orchestration, Governance, Connectors — against the first use case, then expand.

RETAIN

Embedded

We retain a small team alongside yours, owning the substrate while your operators own the work that runs on it.

08 · CLOSE

Build the substrate.

The Operating Layer is what operationally complex organisations build when AI alone is not enough. The case shown here is one of many. The architecture is the same. The substrate is the same. Tellefsen builds it for the operator and hands them the keys.

NEXT STEP

Start with a Sovereign AI Sprint.

Fixed price · 8–12 weeks · Operating Layer Map signed off.

Book the conversation: calendar.notion.so/meet/christoffertellefsen