

Explicit Grounds

The work that comes before the system

Jacob P Slot

Explicit Grounds

On Ray Dalio, Jesper Löwgren, and the work that comes before the system

Ray Dalio built one of the world's largest hedge funds on a principle that sounds obvious and is almost never followed.

He decided to make the grounds for every decision explicit.

Not as policy. Not as culture. As structure. He wrote down the criteria. He expressed them as algorithms. He ran his own reasoning and the algorithmic system side by side and let them interrogate each other. When reality proved a criterion incomplete, he updated it. The decision did not just happen. The grounds for the decision were visible, traceable, and improvable over time.

Dalio called it principles. What he was really doing was explicit grounds — making the foundation of every move readable, not just to himself but to the system that would act on it.

That was thirty years ago. The domain was investment decisions. The system was human.

Jesper Löwgren is asking the same question in 2026. The domain is agentic AI. The system is a loop.

He describes a better loop. Not Think, Act, Observe, Repeat. Something more demanding: State, Evidence, Authority, Action, Proof, Next state. A loop that does not ask what to do next but whether the next move is permitted. A loop that knows when it is allowed to continue, when it must pause, when it must prove, when it must involve a human, and when it must stop.

It is the right description. And it carries the same assumption Dalio carried — that the grounds exist. That the state is readable. That the evidence is structured. That the authority is defined. That the proof can be produced.

Dalio made that assumption explicit at Bridgewater by building it deliberately. He did not assume the grounds were there. He constructed them. Criterion by criterion. Algorithm by algorithm. Updated by reality, not by convenience.

Most organisations deploying agentic AI today have not done that work. They have a loop. They do not have grounds.

This is the gap between the two conversations.

Dalio solved it for a specific organisation, a specific domain, a specific set of decisions. He built the grounds for Bridgewater. They were not transferable as a system, only as a principle. Every organisation that read his book understood the idea. Very few built the structure.

Löwgren describes what the loop needs without describing how the grounds get built. He is right that the loop requires explicit state, defined authority, structured evidence, and governed endpoints. He does not say what making those things explicit actually requires of the organisation before the loop runs.

That is the question neither of them fully answers.

What does it mean to make the grounds explicit for an entire organisation — not just for a trading desk, not just for a claims process, not just for a single agent — but for the operational reality of the organisation itself? Its state. Its permissions. Its tensions. Its boundaries. Its proof.

That is not a governance question. It is not a technology question. It is a structural question. It asks the organisation to do what Dalio did for his decisions: make the criteria explicit, make them readable, make them updatable by reality rather than by assumption.

Now bring that question into a real organisation. Not Bridgewater. Not a thought experiment. The organisation you work in, or work with, or lead.

It has agents running. Some in production, some in pilot, some approved last quarter before anyone asked the hard questions. The demos were convincing. The outputs looked right. The loop is moving.

Ask yourself what the loop is reading.

Not what data it has access to. What it reads as the current state of the organisation. Whether that state was made explicit before the loop ran, or whether it accumulated from models that were built when someone had time, updated when someone noticed, and assumed to be current because nothing broke visibly.

Ask what the loop reads as permitted. Whether the authority structure exists in a form the loop can use at the moment of a transition, or whether it exists in a policy document that a human interprets after the fact.

Ask what the loop leaves as proof. Whether the record of what it did is traceable in the sense that it can be examined, challenged, and owned — or whether it is stored in the sense that it can be retrieved if someone knows where to look.

Most organisations that answer these questions honestly arrive at the same place. The loop is running on implicit grounds. The state was never made explicit. The permissions were assumed. The proof is a log, not a traceable record.

That is not a failure of the technology. The technology is doing exactly what it was built to do. It is a failure of sequence. The grounds were never built because nobody defined what building them meant.

This is where the EA architect makes a different kind of contribution than the one the job description suggests.

Not frameworks. Not principles documents. Not governance layers added to a running system.

The contribution is structural. It is the work of making the organisational reality explicit before the loop runs. Defining the current state in a form that is traceable to its sources. Mapping the permitted transitions in a form that a system can read before it encounters a situation. Creating the record structure that makes proof possible, not just storage. Declaring the human boundaries before the loop reaches them. Defining the governed endpoints before the system needs to stop.

That work is unglamorous. It does not demo well. It does not produce a slide that wins a budget.

But it is the only work that makes everything else trustworthy. The agent that operates on explicit grounds behaves differently from the agent that operates on a model. Not because its reasoning is better. Because what it reads is real.

The compliance officer who has built explicit grounds can answer the regulator's question. Not with a policy document. With a traceable path. Here is the state the system was in. Here is what it was permitted to do. Here is what it did. Here is the decision that authorised it. Here is the proof.

The CEO who has built explicit grounds can stand behind the conclusions the system produces. Not because the system is infallible. Because the path to the conclusion can be examined. Because if something went wrong, the grounds show where and why.

The organisation that has built explicit grounds is not slower than the organisation that has not. It is different. It moves with a connection to purpose, permission, and proof that the ungoverned organisation does not have.

Dalio grounded his decisions. Löwgren grounds his loop. The organisation that wants to move with responsibility has to ground its reality.

Not model it. Not document it. Ground it.

Explicit grounds means the current state of the organisation is traceable to its sources. The permissions are defined in a form that a system can read before it encounters a situation. The tensions are visible as conditions, not discovered as exceptions. The record of every step exists in a form that can be examined, not just retrieved. The boundaries between system decision and human decision are declared before the loop reaches them.

Those are not five requirements for a better agent. They are five properties of an organisation that knows what it is doing and why it is permitted to do it.

Dalio built that for Bridgewater over decades. He called it radical transparency. What he meant was: no implicit grounds. Everything that a decision stands on must be visible to the system that makes it.

Löwgren calls it a governed loop. What he means is: no ungoverned transitions. Every move the system makes must be permitted by something explicit.

The word that connects them is the same word.

Grounds.

The work is not in the agent. The work is not in the loop. The work is in making the grounds explicit before anything moves.

That work has a shape. It has a sequence. It has properties that can be verified. And it is the precondition for everything that Dalio built and everything that Löwgren describes.

Some organisations will build agents first and search for grounds later. Others will understand that the grounds are the work and that everything else follows from them.

The difference between those two paths is not speed. It is not capability. It is not even governance in the conventional sense.

It is whether the organisation can answer, at any moment, for any transition the loop has made: on what grounds.

Jacob P Slot is the creator of ClaritasZ.