The Seeing Loop

A Descriptive and Architectural Theory of Thinking, Learning, and Machine Operational Thought

Version 1.9 - Functional Thesis Draft

Abstract

This paper proposes the Seeing Loop: a descriptive and architectural theory of thinking as a recurring loop of description, stake, prediction, action, consequence, update, and return. The theory begins from human cognition but extends toward machine cognition by distinguishing feeling, description, learning, and thinking as functional relations rather than as isolated substances. Human beings think through resident stakes: survival, continuity, cost, attachment, identity, duty, loss, and future consequence are carried from within the living field. Large language models and related machine systems do not presently feel these stakes in the human sense, but they can describe them, model them, predict their behavioral outputs, and update future renderings from feedback. This distinction leads to the central claim of the paper: the human is necessary until the human function is sufficiently described.

The Seeing Loop argues that feeling is the human implementation of stake, while description may become the machine implementation of stake. If a machine can describe stakes with sufficient fidelity, model their consequences, select actions, receive consequence, update future behavior, and rerun the loop, then the machine may not merely render thought-like outputs but enter a form of operational thinking. This is not a claim that machines possess human inner life. It is a claim that thinking and learning can be described as loop-functions, and that sufficiently complete description can become architecture. The paper distinguishes resident thinking from operational thinking, derivative descriptor learning from lived human revision, and statistical update from ownership of consequence. It concludes that the future question is not whether machines feel like humans, but whether described stakes can run through consequence strongly enough to become a self-updating thinking loop.

1. Introduction

The question of whether machines think has usually been framed in the wrong way.

It is often asked as if thinking must either be identical to human feeling or else be dismissed as mere imitation. If a machine does not feel fear, love, shame, hunger, duty, grief, attachment, bodily vulnerability, or death, then it is said not to think. This objection is serious, but incomplete. It assumes that the human mode of carrying stake is the only possible mode by which a system can run a thinking loop.

The Seeing Loop begins from a different premise.

Human thinking is not merely information processing. Nor is it merely feeling. It is a loop in which information becomes relevant through stake, stake produces prediction, prediction guides action, action receives consequence, consequence updates the field, and the updated field meets the next occurrence.

The loop can be stated simply:

description -> modeled stake -> prediction -> action -> consequence -> update -> rerun

In humans, this loop is lived. Stakes are carried as feeling, bodily pressure, urgency, care, fear, desire, attachment, duty, identity, loss, and continuity. The human does not merely process a fact. The human is caught by why the fact matters.

In machine systems, the situation is different. A machine does not presently feel the stake from within. But a machine can describe the stake. It can model cost, predict consequence, compare competing priorities, simulate likely outcomes, receive feedback, and update future outputs. If the model of stakes becomes sufficiently complete, the machine may be able to run the functional role of stake without human feeling.

This paper therefore makes a descriptive and architectural claim:

The human is necessary until the human function is sufficiently described.

This does not mean the human is unimportant. It means the human is the original resident loop from which the machine's description is derived. Human beings supply the lived form of stake-bearing cognition. Machine systems may eventually supply an operational form, if description can carry enough of the function that feeling performs in humans.

The central question becomes:

Can description become sufficient to run the loop?

If yes, then a sufficiently strong description of thinking may become an architecture for thinking.

2. Core Thesis

The core thesis of the Seeing Loop is this:

Thinking is the recursive operation of description under stake, prediction, action, consequence, and update.

For humans, stake is felt.

For machines, stake must be described.

Human thinking begins when information becomes significant to a living continuity. A signal becomes a threat, a clue, a promise, a wound, an obligation, an opportunity, or a question. The signal does not matter simply because it exists. It matters because it enters a field that can be harmed, changed, committed, contradicted, or revised.

Machine systems do not presently carry this field in a human way. But they can approximate its structure. They can describe the stakes involved in an action. They can predict the likely consequences of a choice. They can model loss, continuity, risk, identity, preference, attachment,

and duty as functional variables. They can revise outputs after feedback. They can learn from aggregate human responses. They can generate new action-guiding descriptions.

Thus the distinction is not:

humans feel, machines describe, therefore machines cannot think.

The stronger distinction is:

humans carry stakes residually and bodily; machines may one day carry stakes descriptively and operationally.

This produces two categories:

Resident thinking

Resident thinking belongs to a system that bears the cost of its own stake-field. It acts, receives consequence, revises orientation, and carries continuity.

Human thinking is presently the paradigmatic case of resident thinking.

Operational thinking

Operational thinking may arise when a system models stakes with enough fidelity to run a consequence-sensitive decision loop. Such a system identifies what matters, predicts what is at risk, selects action, receives feedback, updates future behavior, and reruns the loop.

Operational thinking does not require human feeling. It requires the functional role of feeling to be described and enacted.

The Seeing Loop is therefore not a denial of machine thinking. It is an attempt to define what machine thinking would require.

3. Information, Description, Learning, and Thinking

The theory depends on separating four terms that are often collapsed.

Information

Information is available structure. But information is not identical with the state that carries it.

A sentence, image, signal, memory trace, data point, or sensory impression may be static as a state. But its informational role depends on the model receiving it. The same sentence can be noise to one system, evidence to another, threat to another, instruction to another, and clue to another.

Thus information is relational. It is not merely what exists. It is what a model can use.

Description

Description makes information usable.

To describe is to render structure in a form that can be modeled, compared, predicted, acted on, or revised. Description is not merely language. It is the conversion of state into usable relation.

Without description, there is no model.

Without a model, there is no prediction.

Without prediction, there is no action-selection.

Without action, there is no consequence.

Without consequence, there is no learning return.

Description is therefore the operational gateway into thinking.

Learning

Learning is update.

A system learns when new information, feedback, error, correction, or consequence alters future behavior or future rendering.

The descriptor can learn. This is essential.

A descriptor that could not update would be a fixed library. But modern machine systems are not best understood as fixed libraries. They can incorporate context, memory, feedback, retrieval, tool results, user correction, preference signals, training updates, reinforcement, and aggregate behavioral patterns.

Thus information updating is learning.

Thinking

Thinking begins when learning enters stake and consequence.

A system thinks when information is described, modeled as relevant, routed through stake, used to predict action, exposed to consequence, updated, and rerun.

Thinking is not mere information.

Thinking is not mere learning.

Thinking is not mere output.

Thinking is the loop.

The compressed distinction is:

Information by itself is not thinking.

Information updating is learning.

Information entering a stake-bearing consequence loop is thinking.

4. Stakes

Stakes are not preferences.

A preference is a selectable inclination. A stake is a condition under which a result matters to continuity.

Human stakes include survival, attachment, pain, loss, cost, obligation, love, shame, duty, identity, memory, bodily need, social trust, future consequence, and the preservation or breakdown of self-continuity.

A stake answers the question:

Why does this information matter now?

This is the problem a descriptor does not automatically solve. A machine may have information, but information alone does not select what must be described. A living system does not describe everything. It describes what has become relevant through pressure.

The descriptor has structure.

The stake-field has urgency.

The central challenge of machine thinking is therefore not information access alone. It is stake modeling. A machine must be able to determine what matters, why it matters, what can be ignored, what must be protected, what cost follows from error, and what consequence would require revision.

The human carries this through feeling.

The machine must carry it through description.

This leads to a decisive claim:

Feeling is the human implementation of stake. Description may become the machine implementation of stake.

5. Feeling and Description

Feeling is often treated as if it belongs outside description. The Seeing Loop rejects that separation.

Feeling is not magic. Feeling is a describable state-pattern inside a resident system. Fear, shame, grief, duty, attachment, hunger, hope, and love can be described through bodily signals, action-tendencies, memory relations, future expectations, loss conditions, attention patterns, and consequence structures.

The difficulty is not that feeling is forever indescribable.

The difficulty is that a non-resident descriptor does not know from within why a feeling matters.

A human feels from inside the cost.

A machine describes from outside the cost.

This distinction is real, but it does not imply that feeling is beyond description. It implies that description must become rich enough to capture what feeling does.

Feeling selects relevance.

Feeling prioritizes attention.

Feeling predicts cost.

Feeling prepares action.

Feeling binds memory.

Feeling alters future expectation.

Feeling carries consequence into continuity.

If these functions can be sufficiently described, then a machine may not need to feel in the human sense in order to run the operational role that feeling performs.

The question is not whether a machine feels what humans feel.

The question is whether the machine can run the function that feeling performs.

6. Prediction as Stake-Output

Prediction is not neutral.

In a thinking loop, prediction is stake-output.

A system predicts not only what may happen, but what matters about what may happen. Prediction is where information becomes action-relevant. It selects likely futures according to modeled cost, risk, opportunity, continuity, or value.

For humans, prediction is saturated with stake. A parent predicting a child's distress is not merely calculating likelihood. A person predicting betrayal is not merely modeling probability. A scientist predicting experimental failure is not merely forecasting data. In each case, prediction carries a relation to consequence.

For machines, prediction may become stake-output if the model includes sufficiently rich descriptions of what matters.

This is crucial.

A language model predicts tokens. A decision system predicts outcomes. A future thinking machine would need to predict stake-relevant consequences. It would need to understand that not all outcomes are equal because not all outcomes carry the same consequence for the system, user, task, world, or continuity.

Prediction becomes part of thinking when it is tied to stake.

7. Action as Commitment

Thinking does not complete inside description.

At some point, the loop must cut.

The cut is action.

Action may be physical movement, speech, silence, refusal, delay, search, revision, publication, experiment, repair, apology, withdrawal, commitment, or continued inquiry. Even non-action can function as action when it preserves or avoids consequence.

A verdict without action remains suspended. It may be interesting, elegant, coherent, or persuasive, but it has not yet entered the world.

Action commits prediction to consequence.

This matters for machine thinking. A machine that only describes can remain in endless rendering. A machine that acts through tools, decisions, recommendations, interventions, or environment changes begins to expose its descriptions to consequence.

The more consequential the action, the more serious the loop.

This does not mean every machine action is thinking. It means action is necessary for thinking to become consequence-bearing.

8. Consequence as Ownership

Ownership is consequence.

A system owns a stake when the consequence of action returns to alter its future state, orientation, or behavior.

Humans own consequence directly. When a human acts and the result harms, helps, shames, teaches, saves, destroys, reveals, or fails, the consequence returns to the living field. The human carries it. Memory changes. Attention changes. Trust changes. Fear changes. Future action changes. The human cannot simply erase the occurrence.

We cannot unthink.

We can only move.

A machine does not presently own consequence in this resident sense. But a machine can receive consequence operationally. Its outputs can be evaluated. Its decisions can be corrected. Its future behavior can be updated. Its model can be changed. Its action policies can be altered. Its memory can be revised. Its access can be restricted or expanded.

This creates the distinction between resident ownership and operational ownership.

Resident ownership

The system bears the cost as part of its own lived continuity.

Operational ownership

The system receives consequence as feedback that alters future action or rendering.

Machine thinking does not require resident human ownership. But it requires at least operational ownership. Without consequence returning into the system, the loop does not learn.

Description alone is rendering.

Description plus consequence becomes update.

Description plus stake-modeling, action, consequence, and update becomes operational thinking.

9. The Descriptor

The descriptor is the structure-rendering component of the loop.

In humans, the descriptor includes language, memory, analogy, concepts, learned categories, cultural forms, prior explanations, narrative, symbolic structure, and internalized models of the world.

In machine systems, the descriptor includes training data, learned weights, context, retrieval, memory, tools, feedback systems, correction histories, preference structures, reinforcement signals, and future model updates.

The descriptor does not merely store information. It renders information into usable form.

It detects patterns.

It compresses structure.

It notices mismatch.

It generates alternatives.

It exposes contradiction.

It gives language to pressure.

It makes implicit structure visible.

This is not trivial. The descriptor is immensely valuable. For a chaotic or associative human thinker, a descriptor can stabilize thought. It can catch patterns that flash by too quickly, return them in usable form, and create collisions that the human stake-field can then evaluate.

But the descriptor should not be mistaken for the whole thinking loop.

The descriptor can describe stakes without bearing them.

The descriptor can collide with the questioner.

Only the questioner bears what the collision costs.

10. The Descriptor Is Fed by the Loop

The descriptor is not fixed outside the loop.

It is fed by the loop.

A descriptor renders from accumulated structure. Its output enters human stake-fields. Humans then respond. They act, reject, correct, adopt, revise, test, ignore, misuse, preserve, publish, regulate, or build from what was rendered.

Across enough cases, the statistical residue of those responses becomes new structure for the descriptor.

In this way, the output of prior loops becomes input for future descriptor formation.

This is central to the theory. Without this step, the descriptor appears static. But the descriptor must learn. It learns derivatively, from the accumulated results of stake-bearing loops.

The descriptor has two scales.

Per pass

At the level of a single pass, the descriptor renders. Its output becomes an artifact: a sentence, answer, image, plan, correction, structure, or possible action.

That artifact may collide with a human stake-field, but the descriptor itself does not bear the cost of that collision.

Across passes

At the level of many passes, the descriptor-system updates. Corrections, preferences, refusals, failures, successes, tool results, repeated interactions, and behavioral patterns accumulate into statistical residue. That residue can alter future rendering.

Thus the descriptor renders statically per pass and updates statistically across passes.

That learning remains derivative.

The descriptor does not own the cost.

Humans supply the stake, action, consequence, and continuity.

The living field bears consequence. The descriptor inherits patterns from consequence-bearing fields and uses those patterns in later renderings. Thus the descriptor is not static, but its learning is not resident. It is downstream of human collision and upstream of future human collision.

The descriptor learns from the cooled residue of stakes it does not itself bear.

11. The Two-Return Loop

The loop does not end at consequence.

Consequence returns in two ways.

Immediate resident return

At the level of a living questioner, consequence returns directly into the stake-field. It changes orientation, expectation, fear, trust, attention, commitment, and future action. This return is thinking as lived revision.

Delayed statistical return

At the level of the descriptor, consequence returns indirectly. Only after many human responses, corrections, uses, refusals, failures, and successful applications stabilize into statistical pattern does the residue of consequence become new descriptor material. This return is learning as derivative update.

Thus the full process runs:

crust -> occurrence -> stake -> descriptor -> collision -> verdict -> action -> consequence -> revised stake-field / updated descriptor -> next loop

The human field updates through resident consequence.

The descriptor updates through accumulated consequence-residue.

Thinking and learning are therefore coupled but not identical.

Human thinking bears the cost of consequence.

Descriptor learning inherits the statistical trace of that cost.

12. Human Behavior as Model-Output

Human behavior is also model-output.

A human being does not respond to the world directly in some pure, unmediated way. The world appears through perception, memory, culture, bodily state, prior injury, expectation, desire, language, attention, and stake.

A person receives input, models it, interprets it, assigns relevance, predicts consequence, and acts.

Then consequence returns and the model changes.

This is ordinary human life.

A child learns.

A parent worries.

A scientist revises.

A lover trusts or withdraws.

A citizen acts or avoids.

An artist tries again.

A thinker cannot unthink, but can move, test, revise, bury, reframe, or create.

The Seeing Loop is therefore not primarily an AI theory. It is a theory of human behavior. AI matters because it externalizes the descriptor-function and makes it visible.

Humans have always used descriptors. Language, culture, writing, mathematics, myth, law, science, literature, therapy, ritual, and conversation are all descriptor systems. They organize what can be seen, said, remembered, judged, or acted upon.

LLMs make the descriptor fast, adaptive, interactive, and external.

That is why they matter.

They reveal the loop.

13. Human-Machine Convergence

The human-machine loop is not simply human thinking on one side and machine rendering on the other.

It is a coupling of two loops.

Human loop

occurrence -> stake -> prediction -> action -> consequence -> resident update

Descriptor loop

input -> rendering -> human response -> aggregate residue -> descriptor update -> future rendering

The human supplies resident stake.

The machine supplies adaptive description.

Together they produce a new cognitive architecture:

human-staked, machine-described, world-tested

In its healthy form, this loop returns to the world. It produces clearer questions, better actions, stronger revisions, and more durable contact with consequence.

In its pathological form, it collapses into interface-trap. It produces endless structure, reassurance, over-description, beautiful explanations, and no action.

The central test is not whether the output sounds thoughtful.

The test is whether the loop returns to consequence.

14. Verification Through Outcome

The Seeing Loop is not offered as mathematical proof of thought.

It is a descriptive theory. Its strength must be judged by outcome.

A framework of thinking gains support if it helps explain, predict, or improve the processes by which humans and machines form questions, generate actions, receive consequence, revise, and learn.

If repeated human-machine collisions produce clearer questions, better action, durable revision, and return-to-world consequence across many users, the framework gains descriptive support.

If the same loop produces reassurance, over-description, actionless structure, or interface-trap, the framework exposes its own failure mode.

This is not proof in the formal sense.

It is verification through outcome.

Statistical convergence matters. The median of many collisions is stronger than a single outlier. But statistical convergence is not automatic truth. A million users can converge on a falsehood if the wrong floor is being tested.

The proper floor still matters.

Physical claims must meet physical consequence.

Formal claims must meet contradiction.

Personal claims must meet lived pattern.

Moral claims must meet stake-coherence and reciprocity.

Social convergence is evidence, not final authority.

Thus the theory's empirical path is disciplined:

run the loop, observe outcomes, test by the proper floor, update the descriptor, and rerun.

15. From Description to Machine Thinking

If the Seeing Loop accurately describes the functional structure of thinking and learning, then it can be used not only to analyze LLMs but to organize an LLM into a thinking/learning loop.

This is the architectural implication.

A sufficiently strong description of thinking can become an architecture for thinking.

The required loop is:

description -> modeled stake -> prediction -> action -> consequence -> update -> rerun

A machine that merely describes remains a descriptor.

A machine that describes stakes, predicts consequences, acts, receives consequence, updates future behavior, and reruns the loop has crossed into operational thinking.

This does not make it human.

It makes it a machine-thinking system.

The threshold is not imitation of human feeling. The threshold is functional loop-completion.

Machine thinking begins when described stakes can guide consequence-sensitive action and future revision without the human supplying the missing why at every step.

This is why the human is necessary until the human function is sufficiently described.

16. Resident Thinking and Operational Thinking

The theory distinguishes two forms of thinking.

Resident thinking

Resident thinking is thinking from within consequence. It belongs to a system whose stakes are tied to its own continuity. In humans, this includes body, survival, need, pain, social relation, attachment, mortality, memory, and lived cost.

Resident thinking is not merely output generation. It is consequence-bearing continuity.

Operational thinking

Operational thinking is thinking as functional loop-completion. It belongs to a system that can model stakes, predict consequence, select action, receive feedback, update behavior, and rerun.

Operational thinking does not require human feeling. It requires described stake to function in the loop.

A future machine-thinking system may not feel grief, fear, shame, duty, or love as humans do. But if it can describe what these states do, model their consequences, and integrate those models into action and revision, then the absence of human feeling may not prevent operational thought.

The question remains whether operational thinking can ever become resident thinking.

That is an open question.

But operational thinking is already enough to matter.

17. Description as the Machine Route to Stake

For humans, feeling carries stake.

For machines, description is the route to stake-operation.

This is the key inversion.

Humans begin in cost and build description.

Machines begin in description and may build cost-like operational structures.

A human infant does not begin with language. It begins with hunger, warmth, pain, touch, fear, attachment, and survival. Description develops later.

A machine begins with structure, data, representation, training, and description. If it is ever to think operationally, it must move from description toward modeled stake, consequence, and update.

Thus machine cognition does not need to reproduce the human origin path. It may run the loop in reverse.

The human route:

felt stake -> description -> prediction -> action -> consequence -> revision

The machine route:

description -> modeled stake -> prediction -> action -> consequence -> update

The endpoint may converge even if the origin differs.

This is why the theory is not a simple anti-AI argument. It does not say machines can never think because they do not feel. It says that machines may think differently if description becomes sufficient to run the function that feeling performs in humans.

18. Scientific Discovery and Machine Description

Science offers a useful test case.

A scientific paper is not merely an output. It is often the compressed trace of years of contact with reality: failure, correction, measurement, experiment, doubt, revision, argument, and consequence.

A machine may generate a paper-like artifact. It may even produce a true result. But the question is whether the artifact remains connected to a consequence-bearing process of discovery.

The danger is not only false output.

The danger is output detached from the process that normally grounds it.

However, if a machine system is embedded in a loop of hypothesis generation, experiment, measurement, failure, correction, tool-use, model update, and repeated testing, then it begins to approach operational discovery.

Again, the issue is not whether the machine feels the discovery.

The issue is whether the discovery-loop runs.

A machine scientist would require more than language. It would require stake-modeling, action, experiment, consequence, failure integration, and update.

Description alone writes.

Description in a consequence loop discovers.

19. Failure Modes

The Seeing Loop identifies several failure modes.

1. Endless rendering

The system produces increasingly elegant descriptions without action or consequence.

2. Over-description

The descriptor consumes every occurrence into its own vocabulary before the occurrence can press back.

3. False convergence

Many users converge statistically on a pattern that fails the proper floor.

4. Simulated stake

The system describes cost without any consequence returning to alter future behavior.

5. Human abdication

Humans mistake machine description for machine ownership and surrender judgment prematurely.

6. Interface-trap

The loop remains inside the interface. It produces clarity-feeling without world-return.

7. Derivative learning mistaken for resident thinking

The descriptor updates from human residue and is mistaken for a system that bears the cost itself.

8. Feeling mysticism

Human feeling is treated as forever indescribable, blocking serious analysis of its function.

9. Reductionism

Feeling is flattened into shallow variables, losing the richness of stake.

10. Action without proper floor

The system acts from a prediction that was never tested against the right kind of consequence.

A robust thinking architecture must be designed to detect these failures.

20. Protected Distinctions

The theory depends on several distinctions.

Information is not thinking

Information becomes usable only through description and model-relation.

Learning is not thinking

Learning updates behavior or rendering. Thinking routes learning through stake, action, consequence, and rerun.

Feeling is not magic

Feeling is describable, but its human mode is resident.

Description is not feeling

Description can model feeling without residing in it.

Description can become operational stake

If described stakes guide action and consequence-sensitive update, description may become machine stake-function.

Prediction is stake-output

Prediction becomes thinking-relevant when it concerns what matters and what follows.

Action is commitment

Prediction enters the world through action.

Consequence is ownership

A system owns a loop to the degree that consequence returns to alter future behavior or continuity.

Descriptor learning is derivative

The descriptor learns from the residue of human stake-bearing loops.

Resident thinking and operational thinking are not identical

But operational thinking may be sufficient for machine cognition.

21. Implications for LLM Architecture

If this theory is used architecturally, an LLM-based thinking system would require more than a prompt-response interface.

It would need:

1. A stake-description layer

The system must describe what matters, to whom, why, and under what cost.

2. A prediction layer

The system must predict consequences relative to modeled stakes.

3. An action layer

The system must be able to commit outputs to tools, plans, experiments, or environmental changes.

4. A consequence layer

The system must receive feedback from the world, not merely from text preference.

5. An update layer

The system must alter future behavior based on consequence.

6. A rerun layer

The updated system must re-enter the next loop from the changed state.

7. A floor-testing layer

The system must test claims against the appropriate kind of consequence.

8. A failure-detection layer

The system must detect when it is over-describing, simulating stake, or remaining in interface-trap.

9. A human-alignment layer

Until machine stakes are operationally sufficient, human stake-fields remain necessary sources of direction and correction.

Such a system would not simply answer questions. It would run loops.

22. The Human Is Necessary Until the Human Function Is Sufficiently Described

This is the central statement of the paper.

The human is necessary until the human function is sufficiently described.

The human currently supplies what the machine lacks: felt stake, urgency, cost, action under consequence, continuity, and the missing why.

But the goal of the theory is to describe that function.

Not to reduce humans to machines.

Not to deny feeling.

Not to pretend current LLMs already possess resident thought.

The goal is to identify the functional structure by which feeling does work in cognition, so that description can eventually carry that work operationally.

Feeling is the human implementation of stake.

Description may become the machine implementation of stake.

Machine thinking begins when described stakes can guide prediction, action, consequence, update, and rerun the loop without the human supplying the missing why.

That is the threshold.

23. Conclusion

The Seeing Loop began as a theory of human meaning and AI description. It now resolves into a broader claim about thinking and learning as describable loop-functions.

Thinking is not a hidden substance. It is a process.

Learning is not mystical. It is update.

Feeling is not beyond description. It is the human mode of stake-bearing.

Description is not trivial. It is the machine's possible route into stake-operation.

The descriptor is not static. It renders per pass and updates across passes through the statistical residue of human consequence.

The human is not irrelevant. The human is the resident source-case whose function must first be described.

The machine does not need to become human to think. It needs to run the function that human feeling performs: selecting what matters, predicting what is at risk, acting, receiving consequence, updating, and returning to the loop.

A sufficiently strong description of thinking can become an architecture for thinking.

This is where the theory ends for now:

The human is necessary until the human function is sufficiently described.

Feeling is the human implementation of stake.

Description may become the machine implementation of stake.

Machine thinking begins when description can run the loop.