

From Prompt to Agency

How AI Agents Reason, Use Tools, Remember, and Act

Ngo Trung Kien

Contents

Copyright and publication notice	1
Introduction: the promise and the boundary	3
Visual frontispiece	4
0.1 Who this book is for	4
0.2 Thesis and falsification condition	4
1 Define Agency Without Hype	7
1.1 In plain terms	7
1.1.1 Chapter contract	7
1.2 The decision at stake	7
1.3 Working model	8
1.3.1 See the chapter in practice	8
1.4 Technical deep dive	10
1.4.1 Build sequence	10
1.4.2 Evidence to retain	10
1.4.3 Tests that can disprove the design	10
1.4.4 Design trade-off	10
1.4.5 Chapter-specific operating protocol	10
1.4.6 Operating workflow	13
1.4.7 Evidence and control graph	14
1.5 Worked composite case file — illustrative, not empirical evidence	15
1.5.1 Scenario trace	16
1.5.2 Artifact handoff	17
1.5.3 Evidence application	17
1.5.4 Transfer test	17
1.6 Measurement and recovery decision	17
1.7 Practice lab	18
1.8 Limits and judgment	18
1.9 Takeaways	19
1.10 Reflection	19
2 Read the Agent Loop	21
2.1 In plain terms	21
2.1.1 Chapter contract	21
2.2 The decision at stake	21
2.3 Working model	22
2.3.1 See the chapter in practice	22
2.4 Technical deep dive	24
2.4.1 Build sequence	24
2.4.2 Evidence to retain	24

2.4.3	Tests that can disprove the design	24
2.4.4	Design trade-off	24
2.5	Trajectory trace example	24
2.5.1	Chapter-specific operating protocol	25
2.5.2	Operating workflow	28
2.5.3	Evidence and control graph	29
2.6	Worked composite case file — illustrative, not empirical evidence	30
2.6.1	Scenario trace	31
2.6.2	Artifact handoff	32
2.6.3	Evidence application	32
2.6.4	Transfer test	32
2.7	Measurement and recovery decision	32
2.8	Practice lab	33
2.9	Limits and judgment	33
2.10	Takeaways	34
2.11	Reflection	34
3	Choose Goals and Success Conditions	35
3.1	In plain terms	35
3.1.1	Chapter contract	35
3.2	The decision at stake	35
3.3	Working model	36
3.3.1	See the chapter in practice	36
3.4	Technical deep dive	38
3.4.1	Build sequence	38
3.4.2	Evidence to retain	38
3.4.3	Tests that can disprove the design	38
3.4.4	Design trade-off	38
3.4.5	Chapter-specific operating protocol	38
3.4.6	Operating workflow	41
3.4.7	Evidence and control graph	42
3.5	Worked composite case file — illustrative, not empirical evidence	43
3.5.1	Scenario trace	44
3.5.2	Artifact handoff	45
3.5.3	Evidence application	45
3.5.4	Transfer test	45
3.6	Measurement and recovery decision	45
3.7	Practice lab	46
3.8	Limits and judgment	46
3.9	Takeaways	47
3.10	Reflection	47
4	Design Tools as Contracts	49
4.1	In plain terms	49
4.1.1	Chapter contract	49
4.2	The decision at stake	49

4.3	Working model	50
4.3.1	See the chapter in practice	50
4.4	Technical deep dive	52
4.4.1	Build sequence	52
4.4.2	Evidence to retain	52
4.4.3	Tests that can disprove the design	52
4.4.4	Design trade-off	52
4.4.5	Chapter-specific operating protocol	52
4.4.6	Operating workflow	55
4.4.7	Evidence and control graph	56
4.5	Worked composite case file — illustrative, not empirical evidence	57
4.5.1	Scenario trace	58
4.5.2	Artifact handoff	59
4.5.3	Evidence application	59
4.5.4	Transfer test	59
4.6	Measurement and recovery decision	59
4.7	Practice lab	60
4.8	Limits and judgment	60
4.9	Takeaways	61
4.10	Reflection	61
5	Build Context Not Prompt Piles	63
5.1	In plain terms	63
5.1.1	Chapter contract	63
5.2	The decision at stake	63
5.3	Working model	64
5.3.1	See the chapter in practice	64
5.4	Technical deep dive	66
5.4.1	Build sequence	66
5.4.2	Evidence to retain	66
5.4.3	Tests that can disprove the design	66
5.4.4	Design trade-off	66
5.4.5	Chapter-specific operating protocol	66
5.4.6	Operating workflow	69
5.4.7	Evidence and control graph	70
5.5	Worked composite case file — illustrative, not empirical evidence	71
5.5.1	Scenario trace	72
5.5.2	Artifact handoff	73
5.5.3	Evidence application	73
5.5.4	Transfer test	73
5.6	Measurement and recovery decision	73
5.7	Practice lab	74
5.8	Limits and judgment	74
5.9	Takeaways	75
5.10	Reflection	75

6 Engineer Working Memory	77
6.1 In plain terms	77
6.1.1 Chapter contract	77
6.2 The decision at stake	77
6.3 Working model	78
6.3.1 See the chapter in practice	78
6.4 Technical deep dive	80
6.4.1 Build sequence	80
6.4.2 Evidence to retain	80
6.4.3 Tests that can disprove the design	80
6.4.4 Design trade-off	80
6.4.5 Chapter-specific operating protocol	80
6.4.6 Operating workflow	83
6.4.7 Evidence and control graph	84
6.5 Worked composite case file — illustrative, not empirical evidence	85
6.5.1 Scenario trace	86
6.5.2 Artifact handoff	87
6.5.3 Evidence application	87
6.5.4 Transfer test	87
6.6 Measurement and recovery decision	87
6.7 Practice lab	88
6.8 Limits and judgment	88
6.9 Takeaways	89
6.10 Reflection	89
Midpoint visual synthesis	90
7 Plan Without Pretending Certainty	91
7.1 In plain terms	91
7.1.1 Chapter contract	91
7.2 The decision at stake	91
7.3 Working model	92
7.3.1 See the chapter in practice	92
7.4 Technical deep dive	94
7.4.1 Build sequence	94
7.4.2 Evidence to retain	94
7.4.3 Tests that can disprove the design	94
7.4.4 Design trade-off	94
7.4.5 Chapter-specific operating protocol	94
7.4.6 Operating workflow	97
7.4.7 Evidence and control graph	98
7.5 Worked composite case file — illustrative, not empirical evidence	99
7.5.1 Scenario trace	100
7.5.2 Artifact handoff	101
7.5.3 Evidence application	101
7.5.4 Transfer test	101
7.6 Measurement and recovery decision	101

7.7	Practice lab	102
7.8	Limits and judgment	102
7.9	Takeaways	103
7.10	Reflection	103
8	Control Action and Side Effects	105
8.1	In plain terms	105
8.1.1	Chapter contract	105
8.2	The decision at stake	105
8.3	Working model	106
8.3.1	See the chapter in practice	106
8.4	Technical deep dive	108
8.4.1	Build sequence	108
8.4.2	Evidence to retain	108
8.4.3	Tests that can disprove the design	108
8.4.4	Design trade-off	108
8.4.5	Chapter-specific operating protocol	108
8.4.6	Operating workflow	111
8.4.7	Evidence and control graph	112
8.5	Worked composite case file — illustrative, not empirical evidence	113
8.5.1	Scenario trace	114
8.5.2	Artifact handoff	115
8.5.3	Evidence application	115
8.5.4	Transfer test	115
8.6	Measurement and recovery decision	116
8.7	Practice lab	116
8.8	Limits and judgment	117
8.9	Takeaways	117
8.10	Reflection	117
9	Add Reflection and Verification	119
9.1	In plain terms	119
9.1.1	Chapter contract	119
9.2	The decision at stake	119
9.3	Working model	120
9.3.1	See the chapter in practice	120
9.4	Technical deep dive	122
9.4.1	Build sequence	122
9.4.2	Evidence to retain	122
9.4.3	Tests that can disprove the design	122
9.4.4	Design trade-off	122
9.4.5	Chapter-specific operating protocol	122
9.4.6	Operating workflow	125
9.4.7	Evidence and control graph	126
9.5	Worked composite case file — illustrative, not empirical evidence	127
9.5.1	Scenario trace	128

9.5.2	Artifact handoff	129
9.5.3	Evidence application	129
9.5.4	Transfer test	129
9.6	Measurement and recovery decision	129
9.7	Practice lab	130
9.8	Limits and judgment	130
9.9	Takeaways	131
9.10	Reflection	131
10	Understand Multi-Agent Coordination	133
10.1	In plain terms	133
10.1.1	Chapter contract	133
10.2	The decision at stake	133
10.3	Working model	134
10.3.1	See the chapter in practice	134
10.4	Technical deep dive	136
10.4.1	Build sequence	136
10.4.2	Evidence to retain	136
10.4.3	Tests that can disprove the design	136
10.4.4	Design trade-off	136
10.4.5	Chapter-specific operating protocol	136
10.4.6	Operating workflow	139
10.4.7	Evidence and control graph	140
10.5	Worked composite case file — illustrative, not empirical evidence	141
10.5.1	Scenario trace	142
10.5.2	Artifact handoff	143
10.5.3	Evidence application	143
10.5.4	Transfer test	143
10.6	Measurement and recovery decision	143
10.7	Practice lab	144
10.8	Limits and judgment	144
10.9	Takeaways	145
10.10	Reflection	145
11	Decide When Not to Use an Agent	147
11.1	In plain terms	147
11.1.1	Chapter contract	147
11.2	The decision at stake	147
11.3	Working model	148
11.3.1	See the chapter in practice	148
11.4	Technical deep dive	150
11.4.1	Build sequence	150
11.4.2	Evidence to retain	150
11.4.3	Tests that can disprove the design	150
11.4.4	Design trade-off	150
11.4.5	Chapter-specific operating protocol	150

11.4.6	Operating workflow	153
11.4.7	Evidence and control graph	154
11.5	Worked composite case file — illustrative, not empirical evidence	155
11.5.1	Scenario trace	156
11.5.2	Artifact handoff	157
11.5.3	Evidence application	157
11.5.4	Transfer test	157
11.6	Measurement and recovery decision	157
11.7	Practice lab	158
11.8	Limits and judgment	158
11.9	Takeaways	159
11.10	Reflection	159
12	Assemble a Bounded Agent Blueprint	161
12.1	In plain terms	161
12.1.1	Chapter contract	161
12.2	The decision at stake	161
12.3	Working model	162
12.3.1	See the chapter in practice	162
12.4	Technical deep dive	164
12.4.1	Build sequence	164
12.4.2	Evidence to retain	164
12.4.3	Tests that can disprove the design	164
12.4.4	Design trade-off	164
12.4.5	Chapter-specific operating protocol	164
12.4.6	Operating workflow	167
12.4.7	Evidence and control graph	168
12.5	Worked composite case file — illustrative, not empirical evidence	169
12.5.1	Scenario trace	170
12.5.2	Artifact handoff	171
12.5.3	Evidence application	171
12.5.4	Transfer test	171
12.6	Measurement and recovery decision	171
12.7	Practice lab	172
12.8	Limits and judgment	172
12.9	Takeaways	173
12.10	Reflection	173
	Closing: keep the evidence visible	175
	A compact implementation workbook	177
	Decision register	178
	Evidence scorecard	179
	Exception log	180
	Ninety-day plan	181
	Control and test matrix	182

Workflow map	183
13 References	185

Define Agency Without Hype

In plain terms

Agency is a control-loop property created by delegated choice and action within an environment. In an agent trajectory from goal to termination, the practical question is **When is a system an agent rather than a model call or fixed workflow?** For technical product managers, analysts, architects, and advanced professionals who need a complete mental model of AI agents, the answer counts only when **Create an agency boundary card that names goal observations choices actions state and stop conditions** produces reviewable evidence for **classification agreement rate** without normalizing **using the word agent as a marketing label instead of an operating definition.**

Chapter contract

DECIDE

When is a system an agent rather than a model call or fixed workflow?

BUILD

Create an agency boundary card that names goal observations choices actions state and stop conditions.

MEASURE

classification agreement rate

STOP OR REDESIGN WHEN

using the word agent as a marketing label instead of an operating definition

The decision at stake

For **Define Agency Without Hype**, the agent owner must decide **when is a system an agent rather than a model call or fixed workflow?** before the delegated task crosses from discussion into consequence for the task owner, operator, and affected user. The minimum decision record is the artifact **Create an agency boundary card that names goal observations choices actions state and stop conditions**, the observable **classification agreement rate**, and the explicit hold condition **using the word agent as a marketing label instead of an operating definition.** If any Chapter 1 field is missing, the safe default is to stop tool

use and return the task with its trace to a human.

The bounded **Define Agency Without Hype** decision advances the book's promise to understand the full agent loop and turn vague agent ideas into bounded designs that can be inspected, tested, and stopped. It does not endorse a product or predetermine the Chapter 1 answer; it creates a record that another person can challenge for every representative trajectory and termination review.

Working model

Read the **Define Agency Without Hype** model as four linked fields. **Outcome:** Agency is a control-loop property created by delegated choice and action within an environment. **Boundary:** the design must make **using the word agent as a marketing label instead of an operating definition** visible rather than convenient. **Evidence:** interpret **classification agreement rate** together with review effort and excluded cases. **Authority:** the agent owner may continue, narrow, escalate, or stop the delegated task. For **Define Agency Without Hype**, Reference [2] from Anthropic, *Effective context engineering for AI agents*, informs a Chapter 1 boundary; it is not evidence that the local method or illustrative target will work.

See the chapter in practice

The **define agency without hype in practice** maps *From Prompt to Agency* Chapter 1 into a field scene before its abstract models begin. In one frame, it holds the question **When is a system an agent rather than a model call or fixed workflow?**, the artifact **Create an agency boundary card that names goal observations choices actions state and stop conditions.**, the proof signal **classification agreement rate**, and the warning **using the word agent as a marketing label instead of an operating definition.**

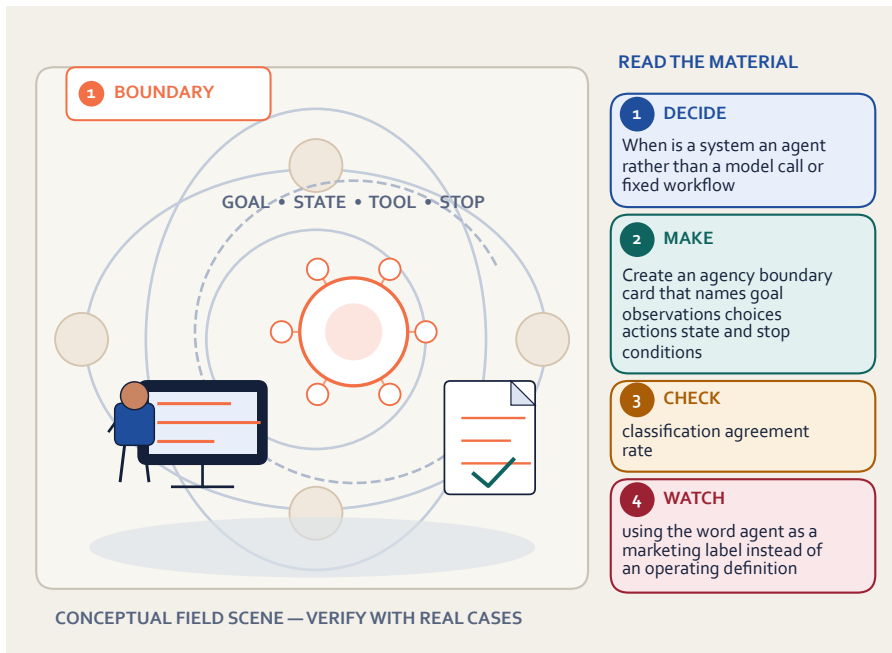


Figure 1.1: Chapter 1 field scene for Define Agency Without Hype: a boundary view in the book’s agent system setting.

Read the scene: For From Prompt to Agency, Chapter 1, begin with the central agent system scene and scan DECIDE, MAKE, CHECK, and WATCH. Treat the Define Agency Without Hype scene as conceptual, then use its cases and evidence records to test whether classification agreement rate actually improves. For **Define Agency Without Hype**, this explanatory plate is a reasoning prompt—not observed evidence or a universal operating prescription.

Visual field note — Define Agency Without Hype:

- 1. Locate the tension.** In this composite situation, A product team calls a retrieval chatbot autonomous even though it cannot choose or act.
- 2. Name the choice.** Answer the chapter question: **When is a system an agent rather than a model call or fixed workflow?**
- 3. Produce the record.** Use **Create an agency boundary card that names goal observations choices actions state and stop conditions.**, then retain the agent owner, assumptions, and case evidence needed to review it.
- 4. Test and return.** Track **classification agreement rate**. If **using the word agent as a marketing label instead of an operating definition**, return to DECIDE and revise the agency boundary instead of explaining the warning away.

Technical deep dive

A model maps input context to an output. A workflow fixes the path in code. An agent receives a goal and can choose among allowed actions after observing intermediate results; that discretionary loop is the practical boundary of agency.

Build sequence

1. List every branch the agent loop may choose at runtime.
2. Mark which branches are fixed by code and which are selected by the model.
3. Name every external state the agent loop can change.
4. Set a maximum loop count and an explicit escalation state.

Evidence to retain

- Agency boundary card with version and owner
- State-transition diagram
- List of reversible and irreversible actions

Tests that can disprove the design

- A task completed by one model call
- A fixed three-step workflow
- A tool-using loop that changes its next action after an error

Design trade-off

More discretion handles variation but increases the space of trajectories that must be evaluated. Within *From Prompt to Agency*, record the rationale for this trade-off, the signal that would reverse it, and the person authorized to decide.

Chapter-specific operating protocol

Phase	Reader action	Evidence retained	Hold condition
1. Define	Write the exact decision behind When is a system an agent rather than a model call or fixed workflow?	Chapter 1 owner, input, output, exception, and timestamp.	Hold Define Agency Without Hype if its record cannot be reconstructed.
2. Contrast	Apply Create an agency boundary card that names goal observations choices actions state and stop conditions to one ordinary case.	Chapter 1 owner, input, output, exception, and timestamp.	Hold Define Agency Without Hype if its record cannot be reconstructed.
3. Exclude	Run the same artifact against a countercase shaped by using the word agent as a marketing label instead of an operating definition.	Chapter 1 owner, input, output, exception, and timestamp.	Hold Define Agency Without Hype if its record cannot be reconstructed.
4. Assign	Name the agent owner as the person who may continue, narrow, or stop.	Chapter 1 owner, input, output, exception, and timestamp.	Hold Define Agency Without Hype if its record cannot be reconstructed.

Phase	Reader action	Evidence retained	Hold condition
5. Challenge	Record classification agreement rate and the review effort for the same case.	Chapter 1 owner, input, output, exception, and timestamp.	Hold Define Agency Without Hype if its record cannot be reconstructed.
6. Publish	Release the Define Agency Without Hype artifact only when its evidence answers this chapter’s question.	Chapter 1 owner, input, output, exception, and timestamp.	Hold Define Agency Without Hype if its record cannot be reconstructed.

This sequence is specific to **Define Agency Without Hype**. For that Chapter 1 decision, it is not permission to automate; it is a way for technical product managers, analysts, architects, and advanced professionals who need a complete mental model of AI agents to make the reasoning inspectable for every representative trajectory and termination review.

Create an agency boundary card that names goal observations choices actions state and stop conditions. This is the chapter’s core production artifact. The **Define Agency Without Hype** artifact should fit on one page, use ordinary language, and be reviewed by someone who did not design it. A compact artifact exposes disagreements that a long narrative can conceal.

The chapter’s visual tool is the **chapter contract**. For **Define Agency Without Hype**, it translates explanation into an inspectable decision aid: the reader can see the decision, artifact, evidence, and exact condition that pauses the delegated task.

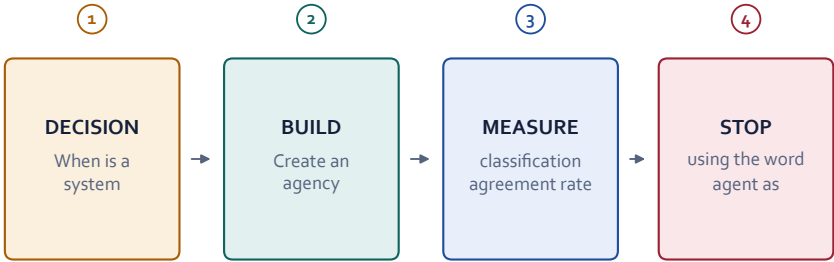


Figure 1.2: Chapter contract connecting the decision, artifact, measure, and stop condition for Define Agency Without Hype.

How to use this page: Move across the four Define Agency Without Hype cards; compare the decision with classification agreement rate and stop before using the word agent as a marketing label instead of an operating definition. The chapter-specific measure is **classification agreement rate**; the principal warning is **using the word agent as a marketing label instead of an operating definition**.

Operating workflow

The **define agency without hype workflow** makes the chapter executable. For **Define Agency Without Hype**, the five stages carry **Create an agency boundary card that names goal observations choices actions state and stop conditions**. from the question **When is a system an agent rather than a model call or fixed workflow?** to a test against **classification agreement rate**, with **using the word agent as a marketing label instead of an operating definition** as an explicit return or stop signal.

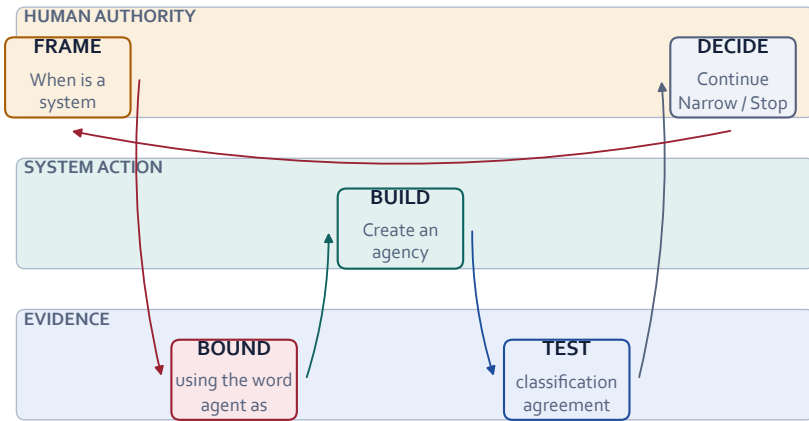


Figure 1.3: Five-stage workflow for chapter 1, Define Agency Without Hype: frame the decision, set the agency boundary, build Create an agency boundary card that names goal observations choices actions state and stop conditions., test against classification agreement rate, then continue, narrow, or stop before using the word agent as a marketing label instead of an operating definition.

Read the workflow: Move from FRAME to DECIDE for Define Agency Without Hype. The lower return path is mandatory when evidence is missing or the stop signal—using the word agent as a marketing label instead of an operating definition—appears. For **Define Agency Without Hype**, record the agent owner and evidence at every handoff; an arrow is not evidence that the preceding stage passed.

Evidence and control graph

The **define agency without hype — evidence network** shows the relationships that must remain inspectable when the workflow runs. In *From Prompt to Agency*, this qualitative graph asks the reader to record observed evidence for **classification agreement rate** and route **using the word agent as a marketing label instead of an operating definition** to recovery instead of treating sample numbers as facts.

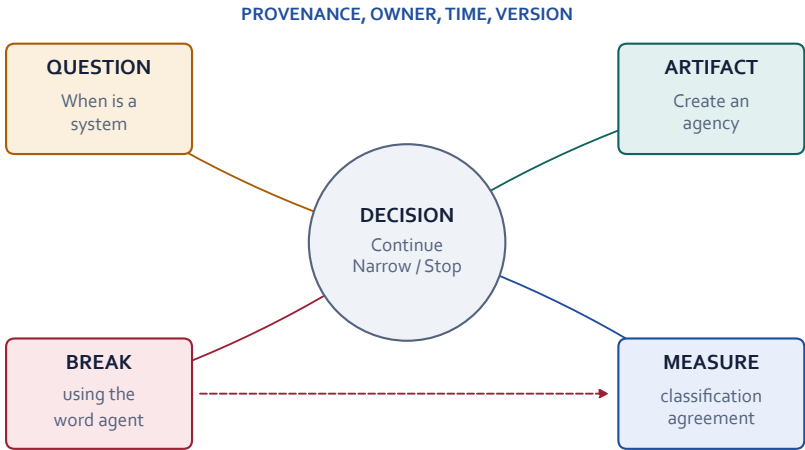


Figure 1.4: Qualitative evidence network for chapter 1, Define Agency Without Hype, linking the question to Create an agency boundary card that names goal observations choices actions state and stop conditions., the measure classification agreement rate, and the recovery condition using the word agent as a marketing label instead of an operating definition without invented empirical values.

Read the graph: Trace the evidence for Define Agency Without Hype toward the decision. For Define Agency Without Hype, empty or weak evidence does not count as zero risk; it routes the delegated task to verification, escalation, or stop. Use this **Define Agency Without Hype** graph beside **Create an agency boundary card that names goal observations choices actions state and stop conditions**, not as a substitute for recorded cases.

Worked composite case file — illustrative, not empirical evidence

This **Define Agency Without Hype** record turns the scenario into an inspectable rehearsal in an agent trajectory from goal to termination. Its numbers are worksheet examples for Chapter 1, not observed results or a prediction of benefit.

Record field	Filled illustrative entry
Situation	A product team calls a retrieval chatbot autonomous even though it cannot choose or act.
People affected	the task owner, operator, and affected user
Decision	When is a system an agent rather than a model call or fixed workflow?
Artifact excerpt	Create an agency boundary card that names goal observations choices actions state and stop conditions.
Accountable authority	The agent owner.
Measure and unit	classification agreement rate — eligible cases meeting the defined condition per 100 cases.
Example baseline	63%.
Example target / decision rule	at least 79%; move only higher, while guardrails remain stable.
Named exception	using the word agent as a marketing label instead of an operating definition.
Safe default	Stop tool use and return the task with its trace to a human.

Scenario trace

1. **Preserve the starting state.** For **Define Agency Without Hype**, record the original request, governing rule, available evidence, and the last safe result before changing anything.
2. **Make the chapter decision.** Answer **When is a system an agent rather than a model call or fixed workflow?** and record the Chapter 1 assumption that carries the most consequence for the task owner, operator, and affected user.
3. **Use the artifact.** Complete **Create an agency boundary card that names goal observations choices actions state and stop conditions** with a normal case and a counterexample; in this chapter, do not infer missing facts merely to finish the form.
4. **Observe the decision rule.** Review **classification agreement rate** for every representative trajectory and termination review; the **Define Agency Without Hype** record reports exclusions and review effort beside the headline value.
5. **Contain and recover.** If **using the word agent as a marketing**

label instead of an operating definition, stop tool use and return the task with its trace to a human. Preserve the Chapter 1 failed case, assign a deadline, and require the agent owner to authorize any restart.

Artifact handoff

Start **Define Agency Without Hype** from the book-level thesis and boundary. Within *From Prompt to Agency*, attach the current artifact and exception record, then hand both to Chapter 2, **Read the Agent Loop**, which asks: What happens between a goal and a completed task?. That recipient may revise the **Define Agency Without Hype** assumption, but must preserve the evidence that made this book's earlier decision reasonable.

Evidence application

Reference [2], Anthropic, *Effective context engineering for AI agents*, provides a relevant boundary or design input for **Define Agency Without Hype**. It does **not** prove that the Chapter 1 worked values, local method, or target will succeed. The **Define Agency Without Hype** reviewer must link this book's exact claim to the exact source passage and leave unsupported synthesis marked UNVERIFIED.

Transfer test

- Re-run **Create an agency boundary card that names goal observations choices actions state and stop conditions** with one routine Chapter 1 case that should pass.
- Re-run the **Define Agency Without Hype** artifact with missing evidence and keep the safe default active.
- Re-run it where **using the word agent as a marketing label instead of an operating definition** appears plausible but not certain.
- Ask a reviewer who did not design this chapter's method to decide whether the retained record supports continuing, narrowing, escalating, or stopping.

Measurement and recovery decision

For **Define Agency Without Hype**, define **classification agreement rate** as eligible cases meeting the defined condition per 100 cases. In this **Define Agency Without Hype** worksheet, the Chapter 1 worked baseline is 63% and the illustrative target is at least 79%; both must be replaced with evidence from an agent trajectory from goal to termination. Review the Chapter 1 result for every representative trajectory and termination review,

preserve excluded and high-consequence cases, and interpret movement as better only when it is higher, while guardrails remain stable.

The named failure is **using the word agent as a marketing label instead of an operating definition**. Containment means: stop tool use and return the task with its trace to a human. Recovery is complete only after the agent owner can reconstruct the failed case, explain which assumption changed, and authorize a new test against **classification agreement rate**.

Practice lab

Choose one live but reversible **Define Agency Without Hype** example from an agent trajectory from goal to termination. Create the following:

- a one-sentence answer to **When is a system an agent rather than a model call or fixed workflow?**;
- a completed first draft of **Create an agency boundary card that names goal observations choices actions state and stop conditions**;
- one routine case, one missing-evidence case, and one case shaped by **using the word agent as a marketing label instead of an operating definition**;
- a locally measured baseline for **classification agreement rate** using the unit eligible cases meeting the defined condition per 100 cases;
- a stop threshold, the agent owner as owner, and the safe default stop tool use and return the task with its trace to a human;
- trajectory evidence packet that will be handed to the next chapter.

Ask a colleague to challenge **Create an agency boundary card that names goal observations choices actions state and stop conditions**. They should identify one ambiguous field, one person among the task owner, operator, and affected user whose interests are missing, and one reason **classification agreement rate** could reward the wrong behavior. Preserve that challenge in the chapter handoff before revising.

Limits and judgment

Define Agency Without Hype supplies a locally testable method, not proof of causal effectiveness or a universal rule. This Chapter 1 method remains inside the book boundary: This book does not claim machine consciousness, autonomous moral agency, universal reasoning ability, or safe deployment without contextual testing. Readers applying **Define Agency Without Hype** for the task owner, operator, and affected user must verify current law, contract, policy, safeguarding, accessibility, security, and professional duties before consequential use. A negative **classification agreement rate** result may mean the question was framed badly, the review

cost is too high, or stop tool use and return the task with its trace to a human is the responsible decision.

Takeaways

- Agency is a control-loop property created by delegated choice and action within an environment.
- The chapter artifact is: create an agency boundary card that names goal observations choices actions state and stop conditions.
- The primary measure is **classification agreement rate**, paired with review cost and failure severity.
- Watch especially for **using the word agent as a marketing label instead of an operating definition**.
- In an agent trajectory from goal to termination, expand scope only after the agent owner can show the evidence and exercise the recovery path.

Reflection

What evidence would let a skeptical reviewer answer **When is a system an agent rather than a model call or fixed workflow?** differently from the agent owner? Which case involving the task owner, operator, and affected user would **classification agreement rate** miss, and would that omission require continuing, narrowing, escalating, or stop tool use and return the task with its trace to a human?

