Artificial Intelligence in a Human World

The AI's job is plausibility, not truth.

The AI's job is plausibility, not truth.

The AI said it compiles, the compiler says no.

Hallucination is the feature, not a bug.

Hallucination is the feature, not a bug.

Unless it is factually wrong... then it's a bug.

Plausible (adj.) – *Content that appears convincing, logical, or reasonable to a human observer, and is truthfully accurate, regardless of its objective truth.*

A controversial statement.

Facts make *statements true*

Facts are Truth-bound

A "statement of fact" is a claim that can be proven true or false, *in this context.*

“Visible Ink”

Beliefs make *stories true*

Plausibile is Story-bound

“Rewrite this email in a friendlier tone *without changing the meaning.*”

“Invisible Ink”

Governance: what the system can trust/act on; *deterministic, judge, receipts*

Epistemics: what the model can say; *storybound, context-limited*

Governance vs. Epistemics

Fact: *Visible Ink*

- **True/False**
- 0 or 1
- Tools
- A Fact
- Determinism

Plausible: *Invisible Ink*

- **Probability**
- 0→1
- Rules
- A Story
- Non-determinism

Governance: what the system can trust/act on; *deterministic, judge, receipts*

Epistemics: what the model can say; *storybound, context-limited*

Governance vs. Epistemics

Fact: *Visible Ink*

- “Is Right” vs. “Sounds right”
- “Is Valid” vs. “Looks valid”
- “Correct” vs. “Convincing”

Plausible : *Invisible Ink*

- In Disney, animals can talk
- In DC, Superman can fly
- In TLOR, the one ring is evil

Governance: what the system can trust/act on; *deterministic, judge, receipts*

Epistemics: what the model can say; *storybound, context-limited*

If a superman movie turned into a batman movie, it wouldn't be a very good superman movie...

The Hallucination Paradox

- **The Feature:** (*Invisible Ink*):
  - When the goal is tone, creativity, or flow, *probability is the engine of utility*.
  - Generating plausible, story-bound content.
- **The Bug:** (*Visible Ink*):
  - When the goal is truth, *probability is a failure of governance*.
  - Generating factually inaccurate claims.

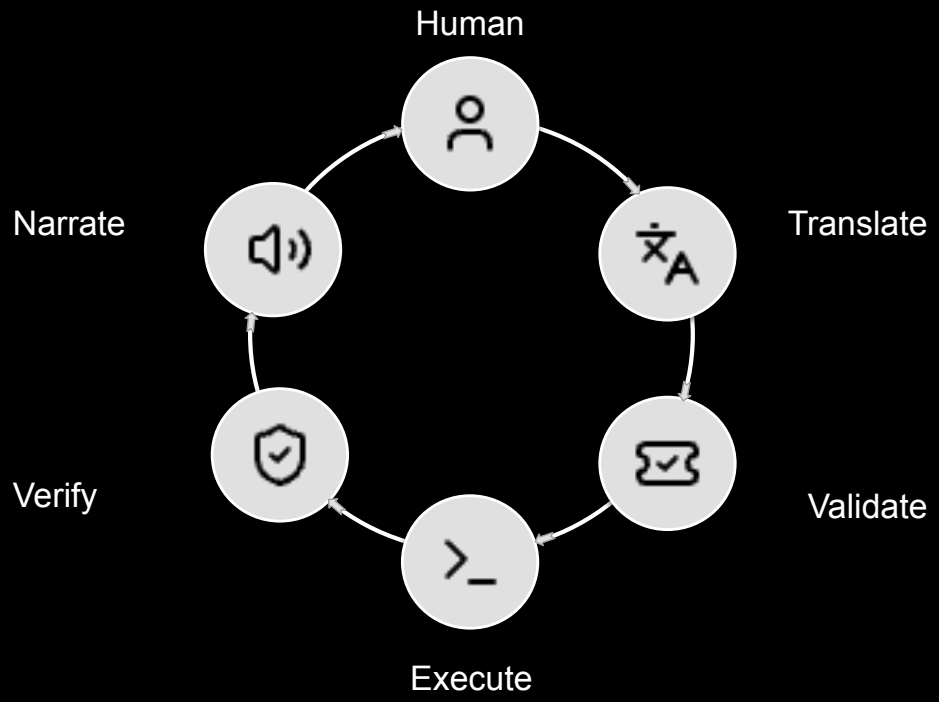
👍 You need both *visible* and *invisible* Ink → Don't fix the model; fix the boundary.

👎 Common Myth: AGI just around the corner and it will fix everything.

The Solver-Checker algorithm

Goals: Protect Compliance and Narrative.

- ✔ Keep the Human in the Loop
- ✔ Align AI/Human Intent
- ✔ Keep AI in the middle
- ✔ Enable Agile AI
- ✔ *AGI is asymptotic to perfect plausibility*

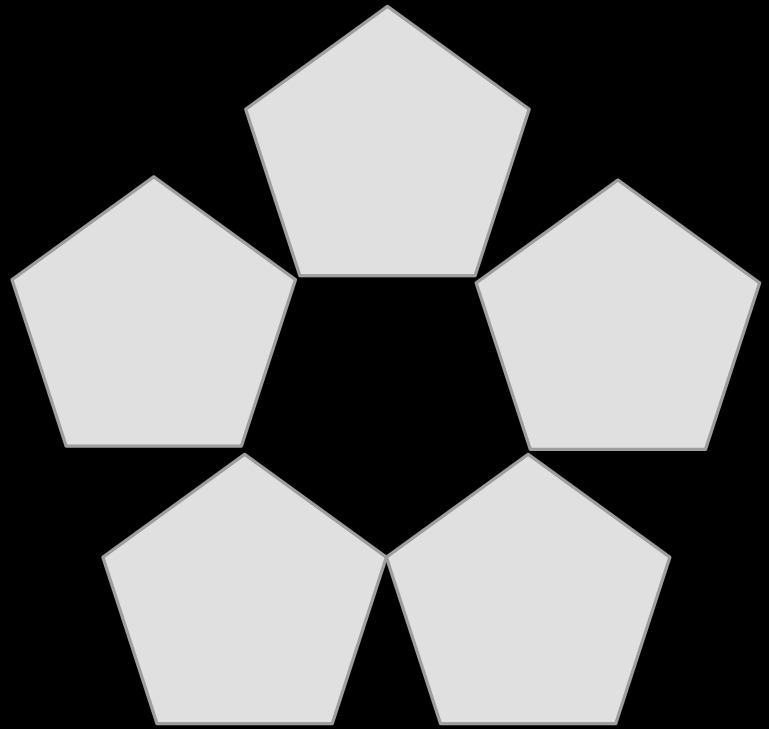


👎 AI Anti-Patterns (common pitfalls in AI development)

1. Single-Shots	*"One prompt, one hope,"* and *"Thou shalt not..."*
2. Models as a solo judge	*Grading your own papers.*
3. No deterministic boundaries	*"No." is a complete sentence. Use it often with AI.*
4. No receipts	*Fire-and-forget vs Proof-of-work. Trust but verify.*
5. Iteration by vibes	*Cargo-cults – Ritual inclusion that serves no purpose.*

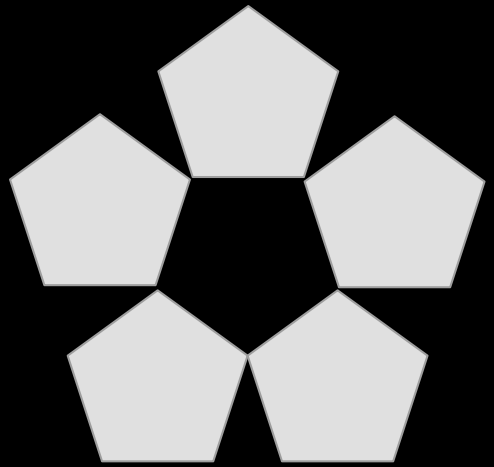
5 Fundamental AI Patterns

1. Grounding
2. Orchestration
3. Verification
4. Trust UX
5. Compound Learning



1. Grounding — "Shared Language" established before execution

WHAT:	Grounding is the anchor
WHEN:	Policies, rules, lore; text-to-source-of-truth
WHY:	Avoid invented facts and GIGO
HOW:	Retrieve anchors + context bundle + execution of tool + knowledge graphs
PROOF:	Citations + “unknown” if missing



👍 Retrieve anchors + citations → say ‘unknown’ if missing

👎 Grounding failures are usually hidden, not obvious

2. Orchestration — coordinating steps in a controlled sequence

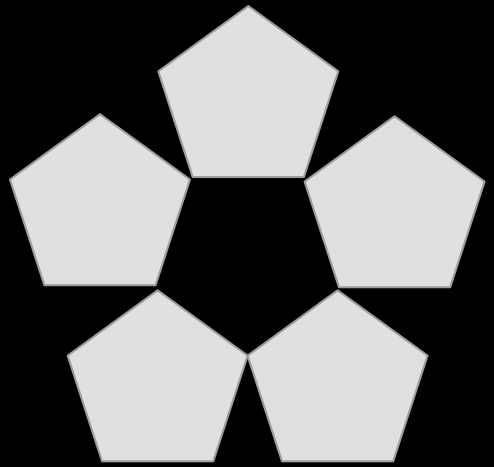
WHAT: Stepwise tool workflow

WHEN: Multi-step procedures

WHY: Prevent skipped/hidden actions

HOW: State machine + token/time budgets + rules

PROOF: "prompt engineering" to "systems engineering."



👍 It is testable, debuggable, and governable → example; accordion editing

👎 Orchestration chains add latency → keep steps minimal and set a latency budget

3. Verification — deterministic checks and proof of work

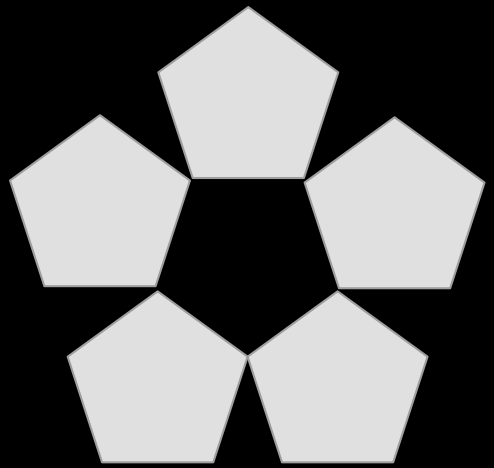
WHAT: Tests decide

WHEN: Binary correctness matters

WHY: Stop plausible wrong outputs

HOW: Deterministic judge boundary

PROOF: Pass/fail receipts

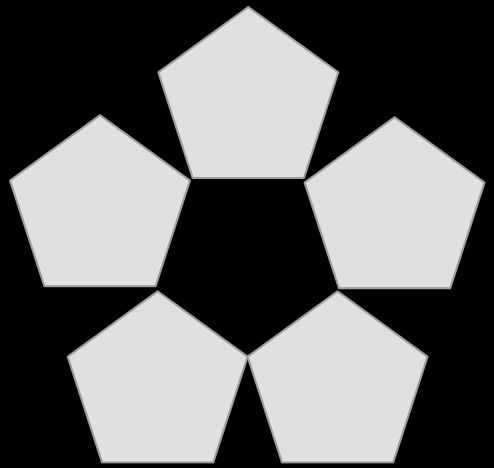


👍 Blind measure model performance objectively → If subjective, use a rubric or reference

👎 Trust vibes. Cherry-picked demos. No pass/fail receipts. Model decides without a judge

4. Trust UX — evidence and recovery options with user in control

WHAT:	Make uncertainty explicit
WHEN:	Users: approve, override, correct
WHY:	Avoid false confidence, (AI's dark triad)
HOW:	Message + receipts; explicit fallbacks
PROOF:	Scope, reasons, choices made, gaps shown



👍 Evidence-based review systems improve trust in decisions

👎 Jagged frontier; silent failures, false confidence, "no receipts, but looks right"

5. Compound Learning — small improvements over time

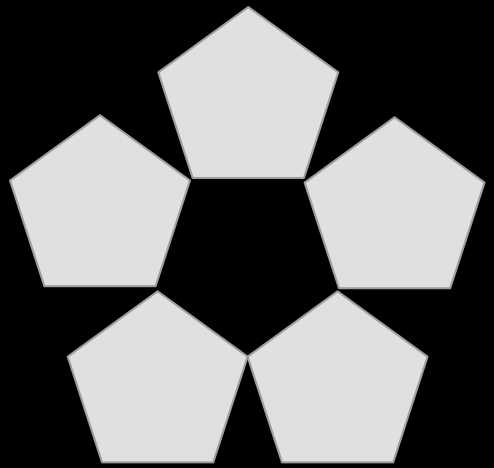
WHAT: Improve without regressions

WHEN: Generate can be scored

WHY: Prevent drift, scale performance

HOW: Offline eval harness + scorecards

PROOF: Score deltas, regression list, diary studies



👍 Self evaluation can drive rapid improvement → *This is far more important than most realize*

👎 Don’t iterate by vibe; use fixed test sets

Solver-Checker Algorithm ↔ Core AI Design Patterns

Algorithmic Step: (what)	**Design Pattern: (how)**	**Reason: (why)**
Translate	1. Grounding	Protects Meaning
Validate	2. Orchestration	Protects Order
Execute	`> Classical Software_`	*Protects The Goal*
Verify	3. Verification	Protects Reality
Narrate	4. Narrate	Protects Honesty
Outside#	*5. Learning*	*Protects Progress*

More on this topic later

👍 Testing in the Loop

- Deterministic unit tests – true unit tests
- Contract tests for the model – unit-test-like, but not text-equality
- Eval regression tests – the real safety net, proof-of-work

- ***Don't unit test creativity – test tools, contracts, and regressions***

👍 AI Testing in the Loop - Practical rules

- Never let the model-judge be the only gate for correctness
- Calibrate the judges with “Golden Sets”
- Prefer pairwise ranking over absolute scoring
- Reduce correlated failure
- Use it to find edge cases
- *A/B tests (Go talk to marketing!)*
- *Employee Synthetic Users*
- In-process testing > fire and then forget (unit tests)

Demo 1: VAWK – AWK vibe coding

Governance: what the system can trust/act on (deterministic + judge + receipts).

1. Grounding:	Backus-Naur Form (BNF) check
2. Orchestration:	Propose → RAG → Run → Patch
3. Verification:	Interpreter + tests decide
4. Trust UX:	Receipts are visible
5. Learning:	Regression sets

https://github.com/dwellman/vawk

Demo 2: A BUUI Adventure

Epistemics: what the model can say (storybound, context-limit - who is allowed to say 'this is correct.')

1. Grounding: World anchored in state transition

2. Orchestration: One command per tick

3. Verification: Game engine-rules decide via RAG

4. Trust UX: State change w/ receipts

5. Learning: Scenario replay with eval

https://github.com/dwellman/adventure

Q&A

The AI's job is plausibility, not truth.

Building the boundary and injecting the facts. That's your job.

(Because, we can't expect foundation models to solve the "truth" problem for us.)

1. **Grounding: AI-assisted triage can notify specialists from imaging workflows.**
   - AI “parallel stroke workflow” tool and workflow timing measures. [AHA Journals]
   - LVO detection software and time-to-treatment/outcomes. [JAMA Network]

2. **Orchestration: AI-assisted stroke triage can notify specialists from imaging workflows.**
   - AI “parallel stroke workflow” tool and workflow timing measures. [AHA Journals]
   - LVO detection software and time-to-treatment/outcomes. [JAMA Network]

3. **Verification: Standardized benchmarks**
   - HELM (multi-metric benchmarking and transparency).
   - BIG-bench (broad task suite; human baselines; calibration discussion). [arXiv:2206.04615]

4. **Trust UX: Evidence-based review systems improve trust in decisions.**
   - Trust in automation review [SAGE Journals]
   - Algorithm aversion [sol3:2466040]
   - The Impact of Placebic Explanations [eiband2019chiea]

5. **Learning: Self evaluation can drive rapid improvement.**
   - Self-Refine: Iterative Refinement with Self-Feedback [arXiv:2303.17651)
   - Reflexion (self-reflection + memory improves agent performance) [arXiv:2303.11366]
   - Constitutional AI (self-critique/-revision framed as AI feedback during training). [arXiv:2212.08]

Presentation Review:

- **Thesis:** The AI's job is plausibility, not truth.
- **Keystone:** AI says it compiles. The compiler says no.
- **Patterns:** 1. Grounding, 2. Orchestration 3. Verification 4. Trust UX 5. Learning
- **Solver-Checker:** Translate → Validate → Execute → Verify → Narrate

Position Papers:

- **Move 37** The shift to reward-seeking behavior
  https://github.com/dwellman/AI/blob/main/papers/move-37.md
- **The Dark Triad of AI** Emergent behavioral risks in self-reinforcing models
  https://github.com/dwellman/AI/blob/main/papers/dark-triad.md
- **Artificial Empathy** Operationalizing ethics through system constraints
  https://github.com/dwellman/AI/blob/main/papers/artificial-empathy.md

Should I Fine tune?

🟧 OUTSOURCE (Do Not Build) High regulation Low feasibility $$$	🟥 PARTNER / CO-BUILD (Shared Control) High regulation High feasibility $$$$$
🟩 AVOID OWNERSHIP (Commodity) Low regulation Low feasibility $$	🟨 BUILD IN-HOUSE (Move Fast) Low regulation High feasibility $$$$