第2世代ループエンジニアリング

自動化の段階モデル

作業を自動化する前に、記録・検出・停止・複数LLM監査・改善を自動化する。

0 Manual から 7 Adaptive へ

- 0 Manual
- 1 Template
- 2 Capture
- 3 Check
- 4 Assist
- 5 Guarded
- 6 Orchestrated
- 7 Adaptive

いきなり自律化しない

第2世代ループの出発点は、AIの稼働時間ではなく監査可能性に置く。

先に自動化しない

作業そのもの
承認判断
不確実性の消去

先に自動化する

記録
分類
抜け漏れ検出
停止条件

合言葉

作業を安全に自動化するための周辺を自動化せよ

八段階の全体像

段階が上がるほど人間が不要になるのではなく、人間の役割が判断へ移る。

段階	名前	自動化するもの
0	Manual	手で作業・記録・判断する
1	Template	Baseline / Risk / Stop を揃える
2	Capture	観測材料を一か所へ集める
3	Check	不足と高リスク信号を呼び出す
4	Assist	Semantic Diff案を下書きする
5	Guarded	フックと権限で止める
6	Orchestrated	実行・検証・複数LLM監査・記録を分ける
7	Adaptive	ループ設計そのものを改善する

段階0-1: 手で回せる形にする

自動化の前に、判断の入口を揃える。

STAGE 0

Manual Loop

目的・非目標・不確実性・試したこと・失敗・次の判断を、人間が手で残す。速度ではなく、何を記録すれば判断が楽になるかを探る段階。

STAGE 1

Template Loop

Baseline、Risk、Stop Conditions、Semantic Diff、Chronicleの空欄をテンプレート化する。判断はまだ人間が行う。

完了条件 空欄を見るだけで、何を確認し、どこで止まるべきかを思い出せる。

段階2: Capture Loop

判断の前に読む観測メモを残す。

変更	実行	確認	信号	介入
ファイル・依存・設定	コマンド・ログ・失敗	テスト・lint・build	送信・削除・権限・秘密	人間の判断・迷い

まだ「安全か」は判断しない。散らばった材料を一か所で読めるようにする。

段階3: Check Loop

承認装置ではなく、呼び鈴を作る。

Info

記録だけする

Warning

確認が必要

Blocker

先に進めない

Critical

テンプレート外

Check Loop は「意味を判断する」のではなく、「意味を判断すべき場面を見つける」。

段階4: Assist Loop

AIは下書き係であり、承認者ではない。

Loop Report Draft

何を実行し、何が未確認か

Semantic Diff Draft

目的・非目標・不確実性の変化候補

Human Review Questions

人間が判断すべき問い

Chronicle Draft

未来へ残す判断の下書き

禁止: approve / safe / 問題なし と断定しない。Risk Levelを勝手に下げない。

段階5: Guarded Loop

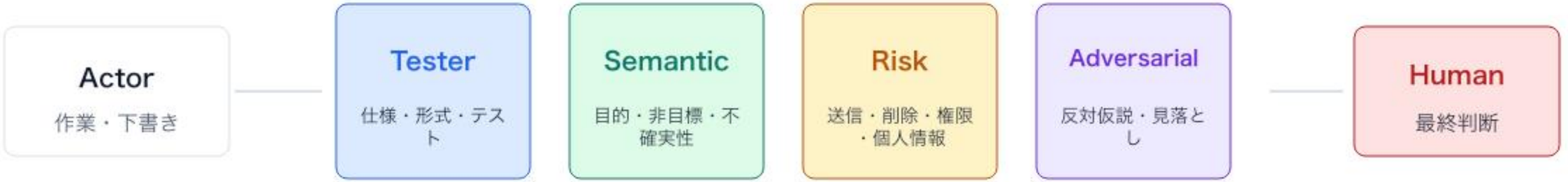
止める仕組みを、判断と記録につなぐ。

Forbidden	**AIに実行させない**	本番DB削除、秘密鍵出力、未承認課金
Pause Required	**人間確認まで止める**	外部送信、権限変更、公開、テスト削除
Auto Continue	**そのまま進めてよい**	低リスクな文言修正、内部メモ更新

再開権限はAIに戻さない。Pause Report、Semantic Diff、人間レビュー、Chronicleを経由して再開する。

段階6: 複数LLM監査

モデルを増やすのではなく、疑い方を分ける。



Consolidated Review は安全判定ではない。

一致した指摘、割れた指摘、一つのLLMだけが出した重大な指摘を残す。

多数決にしない。低リスク判定で高リスク指摘を相殺しない。

段階7: Adaptive Loop

ループ設計そのものを改善対象にする。

IMPROVE

Loop Improvement Proposal

Stop Condition、Risk Tag、Handoff、Semantic Diff、Chronicleの使われ方を読み、改善案を出す。

AUDIT

Meta Chronicle

監査を弱める変更、人間レビューを減らす変更、Guardを軽くする変更の理由と影響を残す。

ループを改善してよい。ただし、ループが何を見なくなったのかを記録せよ。

ツールキット構成

手作業で始め、使われる項目だけを段階的に自動化する。

Start

00 Minimum
01 AI Work Request
02 PR / Delivery

Review

03 Chronicle
04 Human Review
06 Check Report

Capture / Assist

05 Capture Note
07 Assist Draft

Guard / Orchestrate

08 Guard Policy
09 Pause Report
10 Orchestration
11 Role Handoff

Improve

12 Improvement Proposal
13 Meta Chronicle

最小構成は 00 だけ。高リスク領域から 05 以降を足していく。

リスクで摩擦を置く場所を変える

第2世代ループは、すべてを重くする思想ではない。

Low	**軽く回す**	テンプレート + 簡易ログ
Medium	**警告を読む**	Capture + Check
High	**止めて記録する**	複数LLM監査 + Human Review
Critical	**ループ外へ出す**	専門家レビュー + 明示承認

7週間の導入ロードマップ

一気に完全自律へ進めず、周辺の安全機構から足す。

Week 1-4 は判断材料を作る段階。Week 5 以降で停止・複数監査・責任分離へ進む。

避けたい失敗パターン

段階を飛ばすと、速さは出ても途中で何が変わったか見えなくなる。

FAIL 1

完全自律から始める

BaselineもSemantic
Diffもなく、最終出力だけを見ることになる。

FAIL 2

監査を多数決にする

少数の重大指摘を、低リスク判定で相殺してしまう。

FAIL 3

すべてを重くする

LowやMediumまで止め、現場が回避策を探し始める。

FAIL 4

止まった理由を消す

Guardの発火履歴をノイズとして削り、組織の学習を失う。

第1世代ループは、AIに作業を回させた。

第2世代ループは、AIが回る過程を監査可能にする。

人間をループから追い出すのではない。
人間が、より重要な意味判断に集中するための設計である。