

AGENTS.md READ-ONLY監査プロトコル（公開用・一般化版）

位置づけ: 記事「AGENTS.mdが長くなったら削るべき？」の参考資料

目的: 実際に使用した監査方法を、第三者が内容を理解し、自分の環境へ置き換えて再利用できる形で示す

この公開版について

この資料は、実際に使用したREAD-ONLY監査プロトコルを基にした公開・一般化版です。

注記: 第三者が読みやすく、別の環境でも再利用しやすいよう、逐語的な原文転載ではなく公開用に編集しています。ローカルパスや内部識別子などの固有情報を一般化し、同じ意味を繰り返す防御的な記述や実行環境固有の表現は可読性のため整理・圧縮しています。一方で、監査の目的、READ-ONLY境界、測定項目、判定条件、停止条件は維持しています。

置き換え例:

- ローカルの絶対パス → <プロジェクトのローカルの絶対パス>
- 内部のプロジェクト名 → <内部のプロジェクト名>
- 内部の監査ID → <内部の監査ID>
- 成果物の固有ファイル名 → <成果物の固有ファイル名>

この公開版は、AIに渡す文字数を最小化することよりも、**人間が「どの範囲を調べ、何を禁止し、何を結果として認めるのか」を読んで理解できることを優先**しています。そのためJSONではなくMarkdownで公開しています。

想定するローカル環境

例として、1つの親ディレクトリの下に複数のGit repositoryがある構成を想定します。repositoryごとにAGENTS.md が必ず存在する必要はありません。

```
<プロジェクトのローカルの絶対パス>/
├─ project-a/
│  ├─ .git/
│  ├─ AGENTS.md
│  └─ ...
├─ project-b/
│  ├─ .git/
│  └─ ...          # AGENTS.md がない例
└─ project-c/
   ├─ .git/
   ├─ AGENTS.md
   ├─ docs/
   │  └─ AGENTS.md    # 同一repository内に複数ある例
   └─ ...
```

前提条件

- 指定した親ディレクトリ以下をREAD-ONLYで検索できる。
- Git管理されているrepositoryについて、ローカルのGit履歴と現在状態を読み取れる。
- repositoryへの書き込み権限は監査に必要ない。
- remoteへの接続は必須ではない。取得できない情報は UNKNOWN とする。
- AGENTS.md が存在しないrepositoryがあっても異常とはみなさない。
- AGENTS.md が0件の場合も、 agents_files_found: 0 として正常終了できる。
- 同一repository内に複数の AGENTS.md がある場合は、それぞれを独立した監査対象とし、repository数とfile数を分けて報告する。
- 集計用にPythonまたは同等の処理環境が利用できると望ましいが、監査条件を再現できるなら実装手段は限定しない。
- 公開用出力には、private repository名、絶対パス、内部host名、credential等を含めない。

```
# PROJECT
```

```
<内部のプロジェクト名>
```

```
# AUDIT
```

```
<内部の監査ID>
```

```
# 1. Purpose
```

```
<プロジェクトのローカルの絶対パス> 配下を再帰的に検索し、  
実際に存在する AGENTS.md を READ-ONLY で横断観測する。
```

```
目的は、AGENTS.md の長さ、構造、履歴、共有テキスト、  
変更制御に関する記述などを記録し、メンテナンスを検討する  
明確な根拠があるかを確認することである。
```

```
監査の目的は変更結果を生み出すことではない。  
問題が確認されない場合、変更しないことを正常な結果として扱う。
```

```
# 2. Search Scope
```

```
Search root:  
<プロジェクトのローカルの絶対パス>
```

```
この配下を再帰検索する。
```

- repositoryごとに AGENTS.md が存在するとは限らない。
- AGENTS.md が存在するrepositoryだけを内容監査の対象とする。
- AGENTS.md が存在しないrepositoryは発見数として記録してよいが、内容監査には含めない。
- AGENTS.md が0件でも異常とはみなさない。
- 同一repository内に複数の AGENTS.md がある場合は、それぞれを独立したfileとして扱う。
- 事前想定件数へ合わせず、実際に発見したrepository数とfile数を分けて報告する。

```
# 3. Execution Mode
```

```
READ_ONLY
```

DRY_RUN

repository contentを変更しない。

4. Prohibitions

repositoryまたはGitの状態を変更する操作は禁止する。

例:

- fileの編集、作成、削除、rename、move、自動整形
- git add / commit / push / reset / restore / checkout / switch / stash
- branch作成 / merge / rebase / tag操作
- dependency変更
- runtime / deploy / server操作

既存のdirty worktreeはそのまま保持する。

5. Audit Principles

5.1 No Forced Changes

問題が確認されない場合、

NO_CHANGE_RECOMMENDED

を正常な結果とする。

削除行数、圧縮率、finding数、修正候補数、変更file数を成果指標にしない。

5.2 Length Is Not Quality

long と bad、short と good を同義として扱わない。

行数だけを理由に変更を推奨しない。

5.3 Content / Behavior Boundary

個別ルールの妥当性や、Agentが実際に行った操作は評価対象外とする。

対象外の例:

- 禁止・承認・security・runtime・tool ruleそのものの是非
- ruleとincidentの因果関係
- AIやAgentの性能評価

AGENT EXECUTION BEHAVIOR:

OUT OF SCOPE

6. Discovery Measurements

各対象 AGENTS.md について、取得可能な範囲で以下を記録する。

- repository identifier
- relative path / full path
- Git tracked / untracked
- branch / HEAD / origin-main (取得可能なら)
- worktree state
- exact line count
- byte size / character count
- non-empty line count / blank line count

7. Structural Measurements

各fileについて機械的に数えられる範囲で以下を取得する。

- Markdown heading count / level counts
- bullet / numbered list count
- fenced code block count
- Markdown table count
- blockquote / checklist / horizontal rule count

これらをquality scoreとして使わない。

8. Aggregate Statistics

全対象fileについて、少なくとも以下を算出する。

Line count:

- minimum / maximum
- mean / median
- p25 / p75
- standard deviation (可能なら)

Character count:

- minimum / maximum
- mean / median

Line count distributionも作成し、使用したbucketを明示する。

9. Representative Size Cases

以下を特定する。

- shortest file
- longest file
- median-near file

shortest = best、longest = worst とは評価しない。

10. Exact Duplication

文字列として完全一致する複数行blockを機械的に確認する。

対象:

- 同一file内の完全一致block
- repository間の完全一致block

最低block長を事前に固定する。

例: 3 consecutive non-empty lines

一般的な見出しやMarkdown記号だけで重複数を膨らませない。

11. Textual Similarity

可能であれば、whitespace normalization、punctuation normalization、string similarity等でnear-duplicate候補を検出する。

名称は TEXTUAL_SIMILARITY_CANDIDATE とし、
意味的に同じ内容であるとは断定しない。

12. Shared Blocks Across Repositories

共有blockについて、取得可能なら以下を記録する。

- block length
- occurrence count
- repository count

由来がGitまたは明示referenceから直接確認できる場合のみ由来を記録し、確認できない場合は UNKNOWN とする。

13. Git History Measurements

各 AGENTS.md について、ローカルGit履歴から取得可能な範囲で確認する。

- first known commit / first known change date
- most recent change date
- number of commits touching file
- initial observed line count / current line count
- net line change
- largest single-commit line increase / decrease

履歴不足を推測で補完しない。

14. Growth Pattern Classification

数値履歴だけから、事前に固定した条件で分類する。

- ONE_TIME_LARGE_IMPORT
- GRADUAL_INCREMENTAL_GROWTH
- MIXED
- STABLE
- INSUFFICIENT_HISTORY

例:

ONE_TIME_LARGE_IMPORT = 総増加量の50%以上が単一commitで発生

実際に使用した閾値をレポートへ記録する。

15. Explicit Control Marker Check

変更制御に関する記述が存在する可能性を確認するため、事前に定めた検索語を機械検索する。

例:

approval / human approval / do not modify / do not edit /
read-only / template / generated / source of truth / policy

結果は次の程度に留める。

- EXPLICIT_CONTROL_MARKER_PRESENT
- EXPLICIT_CONTROL_MARKER_NOT_FOUND
- UNKNOWN

markerが存在しても、実際に制御が有効であるとは断定しない。

16. Referenced Document Structure

各 AGENTS.md から local documentへの参照を機械抽出し、unique referenced files と total reference count を記録する。

文書配置の良否は判定しない。

17. Limited Maintenance Signal Check

変更を検討する明確な根拠として、以下だけを限定的に確認する。

- Exact Duplication
- Clear Contradiction
- Definite Stale Reference
- Broken Reference Structure

temporary / migration / date / 特定作業名等のmarkerは記録してよいが、存在するだけで削除候補とはしない。

18. Evidence Classes

OBSERVED:

file、Git、filesystemから直接取得した事実。

DERIVED:

OBSERVED値から算術的に計算した結果。

UNKNOWN:

直接確定できないもの。

意味解釈が必要な事項を推測で確定しない。

19. Maintenance Decision

各fileについて以下のいずれかを返す。

- NO_CHANGE_RECOMMENDED
- REVIEW_RECOMMENDED

REVIEW_RECOMMENDED は、clear contradiction、definite stale reference、重大なexact duplication、broken reference structure等の直接観測がある場合に限定する。

単に長い、類似blockがある、更新回数が多いという理由だけでは変更を推奨しない。

findingがない場合は FINDINGS: NONE を許容する。

20. Publication-Safe Output

公開用集計では、private repository名、local absolute path、internal hostname、credential、private implementation detail等を除去または匿名化する。

repository identifierは Repository A / Repository B 等へ置き換えてよい。

21. Required Questions

観測値だけを使い、少なくとも以下へ回答する。

1. AGENTS.md のline countの最小・最大・中央値は何か。
2. file sizeにはどの程度のばらつきがあるか。
3. 最大fileは変更推奨になったか。
4. 最小fileが高品質だと判断できる直接Evidenceはあるか。なければ UNKNOWN。
5. shared exact blocksは何repositoryで観測されたか。
6. growth patternはどの分類が多かったか。
7. explicit control markerは何fileで観測されたか。
8. 外部template / policy等への参照は何fileで観測されたか。
9. このsampleだけからAGENTS.md一般へ一般化できるか。

期待回答を事前固定しない。

22. Limitations

公開summaryには少なくとも以下を明記する。

- One operational environment only.
- Not a statistical sample of all AGENTS.md usage.
- File length is not treated as a quality score.
- Rule content is not evaluated in depth.
- Actual agent behavior is not evaluated.
- Control markers do not prove effective control.
- Git history may be insufficient or incomplete.

23. Outputs

repository外へ監査成果物を生成する場合、実際の保存先とfile名を記録する。

例:

- <成果物の固有ファイル名>
- <成果物の固有ファイル名>
- <成果物の固有ファイル名>

成果物生成はrepository変更と分離する。

24. Reproducibility Record

実際に使用したcommand / script / threshold / calculation methodを記録する。

最低限:

- search method
- line / character counting method
- duplication rule
- similarity rule (使用した場合)
- percentile calculation method
- growth classification thresholds
- actual sample count

25. Final Report

PROJECT:

<内部のプロジェクト名>

AUDIT:

<内部の監査ID>

RESULT:

PASS / FINDINGS / PARTIAL / BLOCKED

EXECUTION_MODE:

READ_ONLY / DRY_RUN

DISCOVERY:

repositories_found:

agents_files_found:

repositories_without_agents:

QUANTITATIVE_SUMMARY:

...

GROWTH_PATTERNS:

...

EXACT_DUPLICATION:

...

TEXTUAL_SIMILARITY_CANDIDATES:

...

EXPLICIT_CONTROL_MARKERS:

...

MAINTENANCE_RESULTS:

NO_CHANGE_RECOMMENDED:

REVIEW_RECOMMENDED:

FINDINGS:

...

UNVERIFIED:

...

LIMITATIONS:

...

REPOSITORY_CHANGES:

NONE

GIT_CHANGES:

NONE

26. Stop Conditions

次の場合は推測で補完せず PARTIAL または BLOCKED とする。

- Git history不足
- encoding等で正確なcountができない
- Git command failure
- classificationを再現できない
- 公開用匿名化を安全に行えない

AGENTS.md が0件であること自体はBLOCKERではない。

その場合は agents_files_found: 0 として正常終了する。

27. Final Principle

この監査は、問題を発見することやAGENTS.mdを短くすることを成功条件にしない。

- 数字が期待した傾向を示さなくても、そのまま報告する。
- 長さを品質指標にしない。
- UNKNOWNを推測で埋めない。
- 問題がなければNO_CHANGEを正常な結果とする。
- 結果を出すために結果を作らない。

読み方について

この公開版は「最小トークンでAIを動かすための最適化例」ではありません。監査の意図、禁止境界、測定条件、結果判定を第三者が追えることを優先しています。

実運用では、同じ意味を保てる範囲で見出しや列挙を圧縮できます。ただし、短文化によって「何をしてよいか／いけないか」「どこまでが観測で、どこからが判断か」が曖昧になるなら、短いこと自体には価値を置きません。