

1. 重大な矛盾/欠落 (P0) :

- ADR承認フローの未定義: Part00とPart14でADRを変更の必須先行と規定しているが、承認者・タイミング・ツール (e.g., GitHub PR承認) の詳細が不明。結果、属人化や承認漏れが発生する可能性。根拠: GitHub ADRガイドライン (<https://adr.github.io/>, 取得日: 2026-01-11) で承認プロセスを推奨。
- Verify自動化の欠落: Part00とPart10で機械判定を必須とするが、CI/CD統合 (e.g., GitHub Actions) が未定義。手動実行依存でVerify忘れが頻発する。根拠: GitHub Actions公式ドキュメント (<https://docs.github.com/en/actions>, 取得日: 2026-01-11) でリポジトリ検証の自動化を標準。
- sources/重複ファイルの扱いが矛盾: Part00で重複削除禁止と明記されているが、Part14で「重複ファイル扱い」の参照が曖昧で、運用時に削除を誘発する可能性。根拠: ADR-0003 (内部参照) と Git公式ベストプラクティス (<https://git-scm.com/docs/git-best-practices>, 取得日: 2026-01-11) で重複管理を明確化推奨。

1. 改善推奨 (P1/P2) :

- ・P1: CI/CD統合の追加: Verifyスクリプト (e.g., checks/verify_repo.ps1) をGitHub Actionsで自動実行。手動依存を減らし、事故ゼロを強化。根拠: GitHub Actionsセキュリティガイド (<https://docs.github.com/en/actions/security-guides>, 取得日: 2026-01-11)。
- ・P1: SBOM生成ツールの標準化: Part01でCycloneDX/SPDXを暫定対応としているが、syft やgrypeのようなツールを明記。2026年時点のサプライチェーンセキュリティ強化に寄与。根拠: CNCF SBOMガイド (<https://www.cncf.io/blog/2023/05/10/software-bill-of-materials-sbom/>, 取得日: 2026-01-11; クロスチェック: NIST SP 800-218 (<https://csrc.nist.gov/publications/detail/sp/800-218/final>, 取得日: 2026-01-11))。
- ・P2: メトリクスツールの導入: Part01でメトリクス計測をSHOULDとするが、Prometheus やGrafanaのようなツールを提案。定期計測を自動化。根拠: Prometheus公式ドキュメント (<https://prometheus.io/docs/introduction/overview/>, 取得日: 2026-01-11)。
- ・P2: 用語集の自動検証: glossary/GLOSSARY.mdの揺れ検出をスクリプト化。Part02で手動確認依存を減らす。根拠: Markdownlint公式 (<https://github.com/DavidAnson/markdownlint>, 取得日: 2026-01-11) で用語チェックのカスタムルール可能。

1. 具体的修正案（Patch案）：

- file: docs/Part00.md
 - change: 11. 未決事項 の U-0001 に追記: 「承認者はリポジトリオーナーとレビュー アー（最低2名）。GitHub PRで承認必須。確認方法: GitHub APIでPR承認ステータスをクエリ。」
 - reason: ADR承認フローの未定義がP0欠落のため、属人化防止と機械判定可能にする。
- file: docs/Part10.md
 - change: 6. 手順 の 手順A: Verify Gateの実行 に追記: 「CI/CDツール（GitHub Actions）で自動実行。ymlファイル: .github/workflows/verify.yml を追加。」
 - reason: Verify自動化の欠落を解消し、再現性を担保。Part00の機械判定原則に準拠。
- file: docs/Part14.md
 - change: 5. ルール の R-1402 に置換: 「ADR先行ルールが定義され、免除条件が明記されている（重複ファイルはタイムスタンプ付きコピーとし、削除禁止）。」
 - reason: sources/重複扱いの矛盾を明確化し、Part00の削除禁止ルールと整合。
- file: docs/Part01.md
 - change: 5. ルール の R-0102 に追記: 「SBOM生成ツール: syft を使用（コマンド: syft dir . > sbom.json）。」
 - reason: SBOMツール未定義をP1改善として具体化し、2026年のセキュリティ基準に適合。
- file: docs/Part02.md
 - change: 8. 機械判定 に V-0201 を追加: 「用語揺れ検出: markdownlint --config .markdownlint.json でカスタムルール実行。」
 - reason: 用語集の自動検証をP2改善として追加し、表記揺れの運用事故を防ぐ。

2. “運用で事故る” 代表シナリオ3つ + 予防策

- ・シナリオ1: ADRなしでdocs/直接変更 → SSOT破壊。予防策: GitHub Actionsでpre-commit hookを導入し、decisions/追加を必須チェック（根拠: pre-commit公式 <https://pre-commit.com/>, 取得日: 2026-01-11）。
- ・シナリオ2: Verify手動忘れて不整合リリース → 事故ゼロ違反。予防策: PRマージ時に GitHub ActionsでFull Verifyを必須とし、FAILでブロック（根拠: GitHub Actionsワークフロー <https://docs.github.com/en/actions/using-workflows>, 取得日: 2026-01-11）。
- ・シナリオ3: sources/誤削除 → 証拠喪失。予防策: Git hookでsources/変更を検知・ブロックし、HumanGate承認必須（根拠: Git hooks公式 <https://git-scm.com/docs/githooks>, 取得日: 2026-01-11）。

1. 未決事項リスト + 確認手順（何をどこで確認するか）

- ・U-0001: ADR承認フロー（承認者リスト・ツール）。確認手順: GitHubリポジトリ設定を確認（<https://github.com/<repo>/settings>）し、PR承認ポリシーをクエリ。必要なら公式ADRツール（<https://github.com/npryce/adr-tools>）を導入検証。
- ・U-0004: Verify自動実行タイミング（commit/push/CI）。確認手順: GitHub Actionsワークフローをテスト作成（<.github/workflows/verify.yml>）し、タイミング別実行ログをevidence/に保存して比較。
- ・U-0101: メトリクス計測頻度（毎日/週次/月次）。確認手順: Prometheusをローカルインストール（<https://prometheus.io/download/>）し、週次/月次ジョブをシミュレート。負荷測定結果をevidence/metrics/に記録。
- ・U-0102: SBOM形式の優先（CycloneDX/SPDX）。確認手順: syftツールで両形式生成（`syft dir . --output cyclonedx-json; syft dir . --output spdx-json`）し、互換性をNISTガイドでクロスチェック。
- ・U-1403: CHANGELOGアーカイブ基準（長大化時）。確認手順: CHANGELOG.mdの行数をスクリプト計測（`wc -l CHANGELOG.md`）し、1000行超で年別分割テスト。結果をdecisions/にADR提案。