

2026.07.24 JaSST'26 Hokkaido



なめらかな「品質」と、その敵

SDAT構造が、エージェントで動き始めた話

ユニファ株式会社 AI開発推進部 QAエンジニア

高田 佑樹

高田佑樹

ユニファ株式会社
プロダクトデベロップメント本部
AI開発推進部 QAエンジニア

略歴

～2015年：レコーディングスタジオのスタッフや家電の販売員
2015年～2019年：SES事業会社で、IT業界デビュー
2019年～2024年9月：ゲームの第三者検証会社にて従事
2024年10月～現在：ユニファ株式会社でQAエンジニアとして従事

趣味：HIIT・自重トレーニング、ドリップコーヒー
(イエメニア種のコーヒー豆を、もう一回飲みたい)

X @tackaaaada



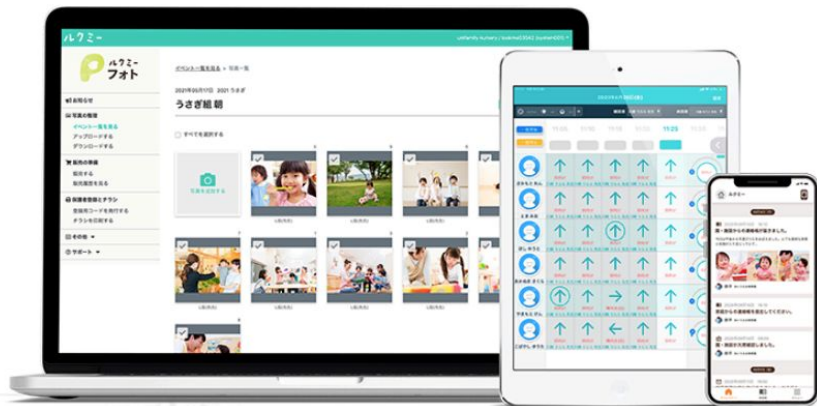
ユニファ株式会社って、どんな会社？

保育施設向けの総合ICTサービス「ルクミー」を提供している会社です

Our Product



ルクミーは、子どもともしっかり向き合える
豊かな環境を整え、
保育AI・写真・ICTで、
選ばれる園づくりを支援する
保育園・幼稚園・こども園向けサービスです。



ユニファ株式会社って、どんな会社？

ルクミーのプロダクト一覧

保育現場の業務負荷軽減



連絡帳や登降園、IoTサービスと連携し、クラス運営に必要な情報を一元管理できます。



指導計画や保育日誌などを監査対応した形でデータ化し、管理できます。



指導計画や保育日誌などを監査対応した形でデータ化し、管理できます。



指導計画や保育日誌などを監査対応した形でデータ化し、管理できます。

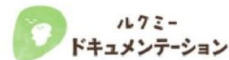
保育の質&保護者コミュニケーション支援



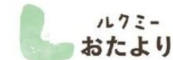
保育施設と保護者の連絡をスムーズにします。



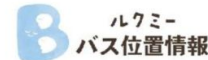
専用アプリで撮影すると、作業が自動化される写真販売サービスです。



ルクミーフォトと連携し、保育の振り返りを手間なく行えます。



おたより・お知らせを保護者へ確実に手間なく一斉送信できます。



送迎バスの位置情報をリアルタイムでお知らせします。



保護者アンケートの一括管理システムです。

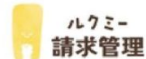
園・施設の運営支援



園児台帳・発達記録・健診・要録を集約。園児情報の一元管理システムです。



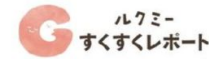
園児の登降園時間を記録・管理します。



延長保育料を自動計算し、請求書の作成も簡単にできます。



シフト作成時の集計チェックを楽にするシフト管理&勤怠管理システムです。



保育AIが過去の写真や日誌を自動整理し、AIレポートを作成。

保育AI

ユニファ株式会社って、どんな会社？

プロダクト×AI①

保育AI™

保育×AIでの課題解決を推進。

先生の負担を大きく軽減するだけでなく、
AIだから気づける子どもの育ちを
提示できるようなソリューションに。

ユニファは、いち早くAIと保育の可能性を
掛け合わせ、保育業界におけるAI活用の
トップランナーを目指しています。



アジェンダ

- 3つの負債と、3つの問題
- じゃあ、どうするの？
- 実践して、どうだったんですか？
- 品質の「○○○○」へ

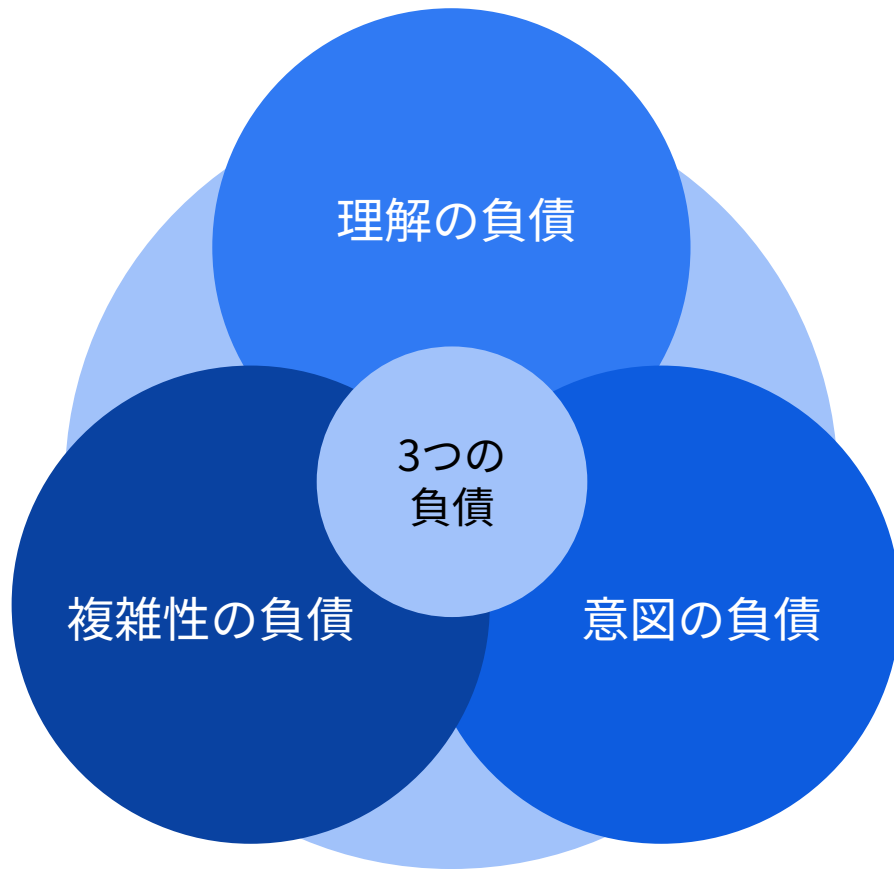


3つの負債と、3つの問題

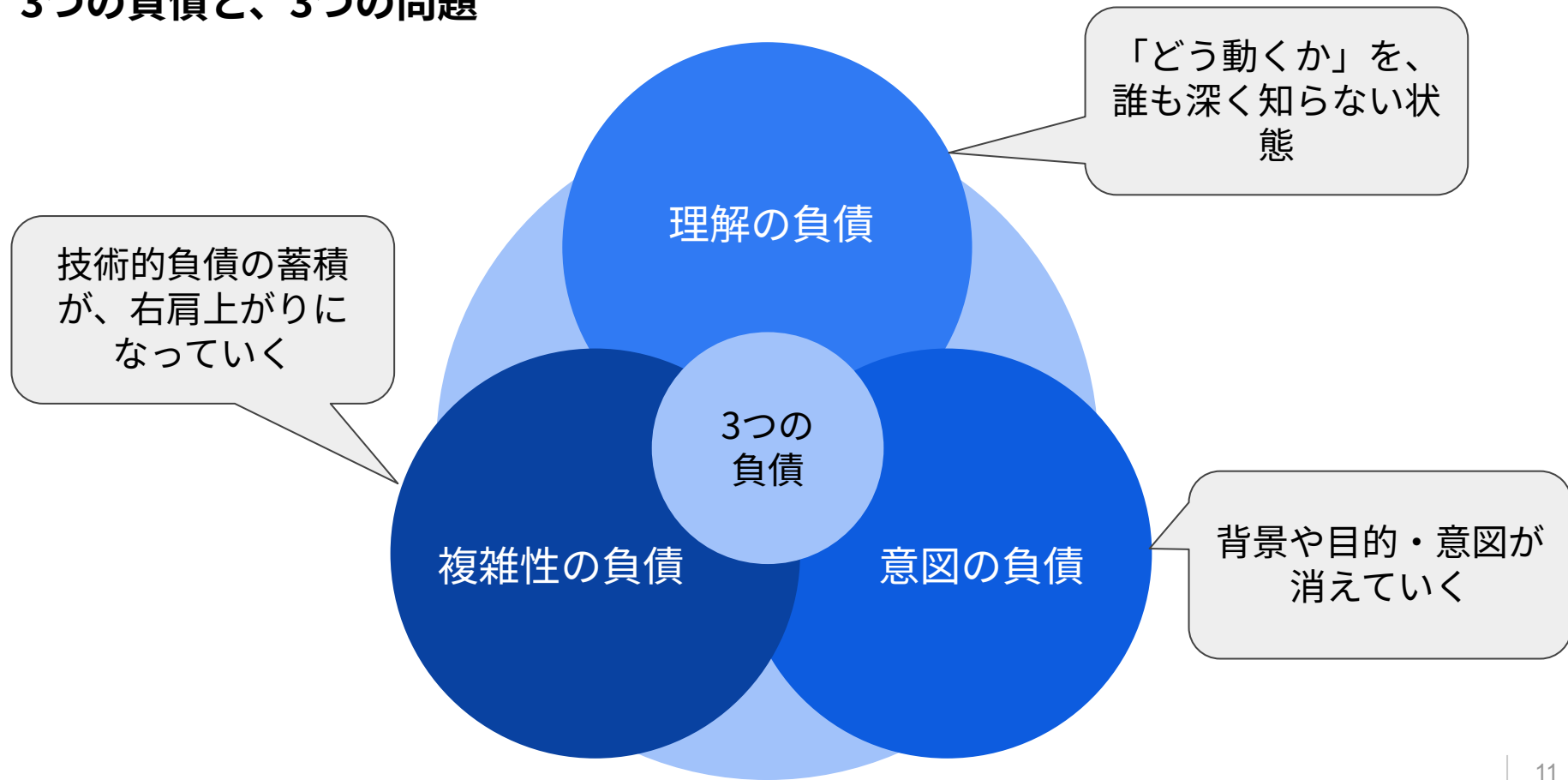
「**意図**」は、伝わってますか？

**AIは「差を広げる」ことによって、
品質を「バグらせる増幅器」だった**

3つの負債と、3つの問題



3つの負債と、3つの問題

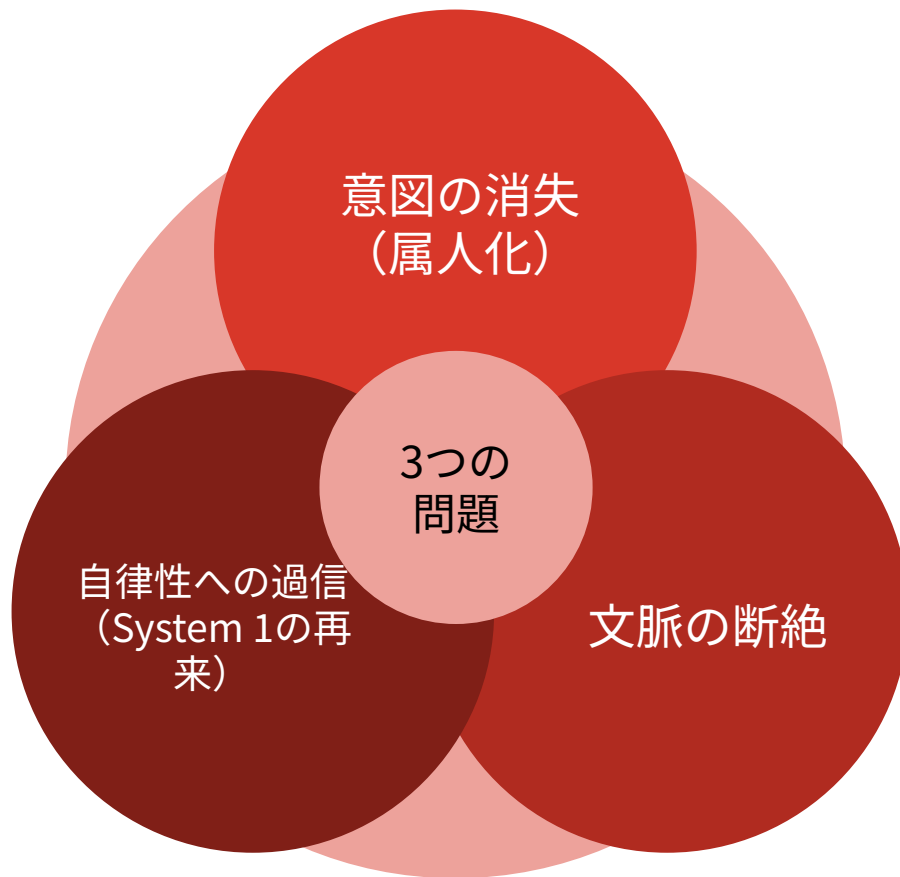


3つの負債と、3つの問題

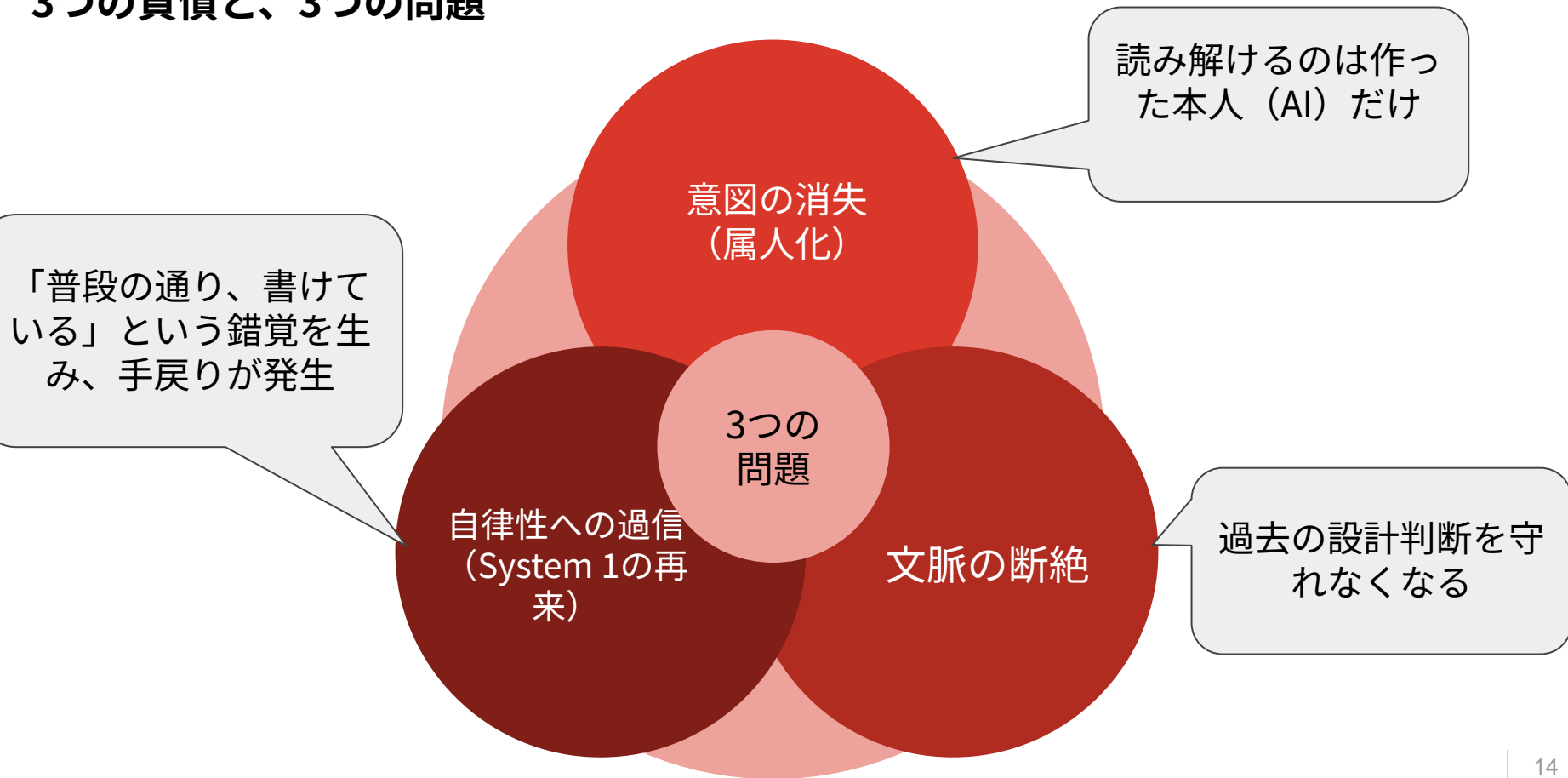
高田が体験した、3つの実例

1. 仕様書をAIエージェントに渡したら、大量のテストケースが作成され、そのテストが「テスト対象/テスト観点/テスト方法」のどれに当たるのか、なぜその対象にしたのか、観点・方法についても同様に、なぜそうしたのかが一切残っていなかった。
2. 新しいセッションで同じプロンプトを使っても、AIエージェントの特性である「確率的に変動する出力」によって違うテスト内容が出てくる。次の日には、また別の内容が出力されて混乱する。（あれ、何のテストをしたかったんだっけ？）
3. エージェントによって各成果物が生成され、『なんか良さそう』と思ってしまったという事実から、「普段の通り、書けている」という錯覚を生んでいた。（その後、テスト観点の漏れが多発）

3つの負債と、3つの問題



3つの負債と、3つの問題



3つの負債と、3つの問題

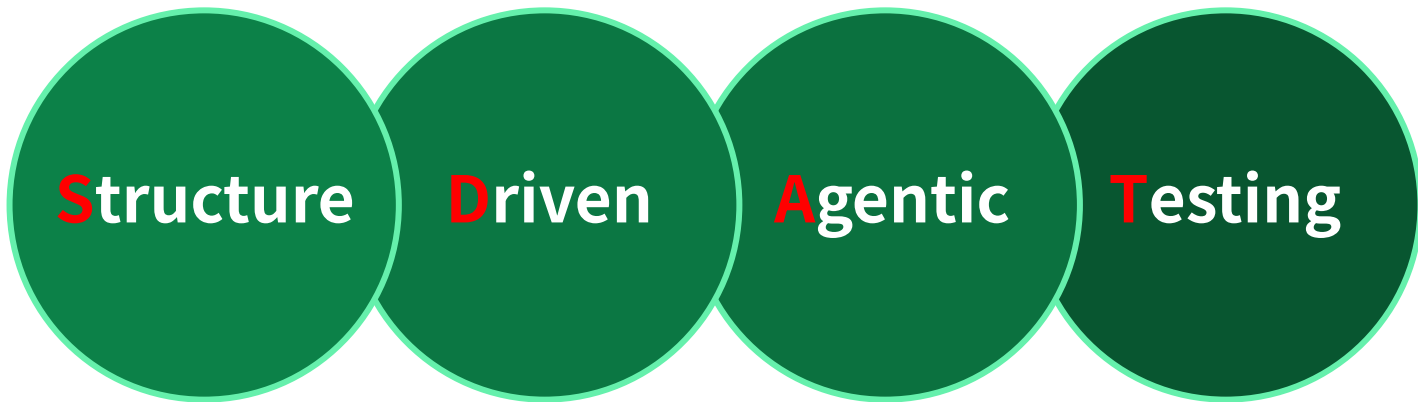
これらの負債・問題を、解決する必要がある。

じゃあ、どうするの？

じゃあ、どうするの？

S D A T

じゃあ、どうするの？



プロンプトではなく、「構造を埋め込む」

じゃあ、どうするの？

「判読性 / 読みやすさ (legibility) 」
×
品質のゴールデントライアングル

じゃあ、どうするの？

「判読性 / 読みやすさ (legibility) 」 (WYSIWID要素)

01

Concepts (概念)

- 独立した状態とアクションを持つ機能単位
- 他の機能と依存しない「個」としての存在

02

Syncs (同期)

- Concepts同士の連携ルール
- あるConceptの状態変化が、別のConceptにどう影響するか

03

Actions (操作)

- Conceptsに対する具体的なメソッド実行
- クリックや入力などの具体的な操作手順

じゃあ、どうするの？

「品質のゴールデントライアングル」

01

Where? (テスト対象)

- どこをテストするのか (テスト対象・評価範囲)
- どの機能、画面、API、データ、状態、境界を確かめるのか

02

What? (テスト観点)

- 対象について何を確かめるのか (テスト観点・品質上の問い)
- どの性質、振る舞い、リスクに注目するのか

03

How? (テスト方法)

- どのようにテストするのか (テスト方法・評価戦略)
- どのテスト技法、テストレベル、ツール、手順で確認するのか

じゃあ、どうするの？

「判読性 / 読みやすさ (legibility)」 (WYSIWID要素) × 「品質のゴールデントライアングル」

01	Where? × Concepts	• テスト対象：どの独立した機能単位をテストするか • 機能、画面、API、データ、状態、境界
02	What? × Syncs	• テスト観点：何のイベント駆動ルールをテストするか • どの性質、振る舞い、リスクに注目
03	How? × Actions	• テスト方法：どのような手順で操作してテストするか • ツール、テストレベル、ツールも考慮
04	Who?	• テストをするにあたって、「誰の視点」でテストをすべきか • 例1) ユーザー：園の先生、園長先生...
05	Why?	• テストPJの目的や背景も、明文化する • 「Why (なぜそのルールが必要か)」と「Time (いつ発火するか)」といった「意図と背景を明文化」する

これに「Then Properties」という「検証可能な性質の定義、振る舞い」を定義する。



実践して、どうだったんですか？

実践して、どうだったんですか？

技術スタック：Claude（Claude Code / Claude Cowork） + Agent Skills

実践して、どうだったんですか？

プロジェクト憲法 (qa-constitution)

Who/Whyを成果物に永続化する

憲法(constitution)をSkillsに定義し、Who/Whyを構造として宣言し、成果物に書き込ませる

「①意図の消失」に対する、対応

Agent Skills (6 Phase・7 Skills)

構造から、セッションをまたいで継承する

Agent Skillsを使って、構造を作る手順をセッションをまたいで引き継ぐ

「②文脈の断絶」に対する、対応

ガードレール + Generative Coverage

人間の批判的思考を構造に組み込む

人間がPropertyを定義し、AIがケースを生成する

「③自律性への過信」に対する、対応

実践して、どうだったんですか？

プロジェクト憲法 (qa-constitution)

Who/Whyを成果物に永続化
する

仕様理解ギャップ由
来のレビュー指摘

0件※

Agent Skills (6 Phase・7 Skills)

構造から、セッションをま
たいで継承する

過去の設計判断の
再説明

insights 10件がSkills
本文へ。還流不要に

ガードレール + Generative Coverage

人間の批判的思考を構造に
組み込む

AI生成観点の削除・
却下

0件
(過剰生成なし)

※：レビュー指摘そのものの差は、中程度。追加ベースでの指摘率：9.1%(適用前)→7.7%(適用後)、指摘の総数ベース：18.2%(適用前)→10.3%(適用後)。また、仕様ギャップ指摘が0件になったのは、Skillsという道具ではなく、qa-specが仕様をConcepts/Syncs/Propertiesに構造化してから設計に入る構造の効果

実践して、どうだったんですか？

▲**59~73%**

工数削減

SDAT適用前
： **9.2~14.2人 / 日**

SDAT適用後：
約3.8人 / 日

77%→94%

シナリオ網羅性

SDAT導入直後は
55%に低下

Insightsの還流
ループで**94%へ**

50%→100%

欠陥検出率(※)

適用前は、設計完了後
に**3件**、テスト実行で
3件の手戻りが発生

適用後は、指摘の
100%が設計段階で
反映

実践して、どうだったんですか？

意図の消失：「憲法：Who/Whyを構造として宣言し、成果物に書き込ませる」

CLAUDE.md
(プロジェクト憲法)



qa-constitution
(SDAT原則)



全7 Skills
の根拠として機能

実践して、どうだったんですか？

意図の消失：「憲法：Who/Whyを構造として宣言し、成果物に書き込ませる」

根 qa-plan 「テスト計画の鉄則」

Who? : ユーザーの導出 (Concepts)
を明確にする

When? : リリース日／リリース期間の
導出

Why? : テスト目的・背景の導出
(Purpose)

What? : 主要な同期シナリオの導出
(Key Syncs)

実 spec.mdの「Who / Why」

Who (Persona / Access Policy) :

- ・園の先生 (先生管理者) :
- ReportItemSettingの操作・レポート作成・
閲覧
- ・園管理者: 項目設定の主担当

Why (Purpose) :

すくすくレポートの利用率向上のため、○○
データをレポートの引用元として追加する。
データの質向上が有用性の実感につながり、
新規利用開始園の増加を狙う。

実践して、どうだったんですか？

文脈の断絶：「構造を作る手順を、セッションをまたいで引き継ぐ」

フェーズ	Skill	成果物
PLAN	qa-plan	テスト計画書 (Who/When/Why/Key Synchs)
DESIGN	qa-design	テスト設計書 ・ Viewpoint Matrix
IMPLEMENT	qa-implement	テストケース + テストデータ
EXEC	qa-exec	実行結果・バグチケット
REPORT	qa-report	テスト完了報告書
EVALUATE	qa-evaluate	周期評価・insights の還流
全体ルール	qa-constitution	プロジェクト憲法

Viewpoint Matrix 列定義

- ID
- 観点
- 対象(Concepts)
- 方法
- 条件(Given)
- 期待結果(Then Property)
- 想定ケース数
- 備考(Who)

実践して、どうだったんですか？

文脈の断絶：「構造を作る手順を、セッションをまたいで引き継ぐ」

10

件の insights が Skills 本文へ還流済み
(standards/insights/)

還流ループの実例

- 新規機能開発 のテスト実施中、「セキュリティ境界は組織間データ漏洩が最重大、組織内ロール制御は次点」と判明
- insights/security_boundary_hierarchy.md として記録
- qa-spec SKILL.md Step 5 に参照リンクとして反映済み
- AIはセッションごとにリセットされ学習しないことから、学習ループを作った。

実践して、どうだったんですか？

ガードレール＋Generative Coverage：「人間の批判的思考を構造に組み込む」

1 Principle of Generative Coverage (qa-design)

人間は Then Property（～であること）を定義し、AIがエッジケースを生成する。
「何が正解か」は人間が決める。

Then Property の記述ルール（重要）

- 単なる「画面が表示されること」で終わらせない
- 「なぜそれが正解か（Why）」が伝わる形にする
- 非機能は測定可能なQuality Propertyとして記述
例：`p95 <= 300ms (50rps, Staging-1環境)`
- ****Principle of Generative Coverage****: 人間はPropertyを定義し、AIがエッジケースを生成する

実践して、どうだったんですか？

ガードレール＋Generative Coverage：「人間の批判的思考を構造に組み込む」

2 Critical Thinking Check ※ (qa-design Step 1)

設計を始める前の検証チェックリスト：

- ・各 Viewpoint に対応する Who / Why が記述できるか
- ・エッジケース・エラー経路が少なくとも1つ含まれる予定か

Step 1: Critical Thinking Check (設計前検証)

設計を始める前に以下を確認する：

- [] FB反映済みのテスト計画書が承認されているか
- [] qa-plan Step 4.5 でこのテスト設計書が属するPR境界（テスト設計書PR）が定義されているか
- [] PJ仕様書・API仕様書が確定しているか
- [] STG環境情報が確定しているか
- [] エッジケース・エラー経路が少なくとも1つ以上含まれる予定か（Step 2.5のChrome DevTools調査を実施したか）
- [] 各Viewpointに対応するWho / Whyが記述できるか

実践して、どうだったんですか？

ガードレール＋Generative Coverage：「人間の批判的思考を構造に組み込む」

3 ガードレール (GUARDRAILS.md)

G-COM-02 承認ゲート遵守：ユーザー承認なしに次の Bolt へ進むことを禁止

G-COM-01 1ターン1問制限 / 不要な質問カテゴリ1～4 (※)

🛡️ ガードレール (作業開始前に必ず確認)

> 詳細: [guardrails.md](./guardrails.md) / 共通: [../guardrails/GUARDRAILS.md](../guardrails/GUARDRAILS.md)

****禁止事項 (この Skill で絶対にしてはいけないこと)**:**

- ****G-D01**:** Then Property なしでテスト観点を作成しない
- ****G-D02**:** 「正常系」「異常系」「エラーケース」等の抽象表現を観点に使わない
- ****G-D03**:** Concepts Definition テーブルなしに Viewpoint Matrix を作成しない
- ****G-D04**:** Concepts間に依存関係を持たせない
- ****G-D05**:** FB反映済みテスト計画書の承認前に設計を開始しない

****禁止事項 (共通) **:** 既存のアーティファクトに書いてある情報を再確認しない / 1ターン2問以上質問しない / ユーザー承認なしにフェーズを進めない

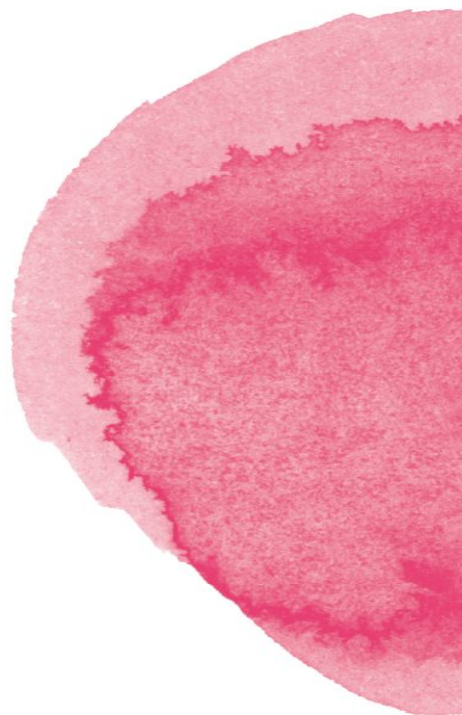
****不明点の扱い (G-COM-08～10) **:**

- 不明点を発見したら「不明点リスト」として出力し、推測で補わない
- 「たかだか知っているもの」は1問ずつ確認 / 「調査が必要なもの」はAIが調査してたかだと共に検証
- 仮定を置く場合は必ず「仮定: ∞と仮定します (要確認)」と明示する

実践して、どうだったんですか？

**「エージェントが動いている＝普段通り書けている」ではなく、
「SDATという、人間が定義した検証条件と行動制約の範囲で」エージェ
ントが動いている状態に変えた。**

品質の「○○○○」へ



品質の「○○○○」へ

AIがテストケースを書く時代に → 品質の「**モデラー**」へ

品質の「モデラー」へ

**構造を設計し、Propertyを定義し、
insightsを還流させる役割は、むしろ重要性を増す**

品質の「モデラー」へ

作業は縮小するが、品質をエンジニアリングすることは無くならない

Appendix

Appendix

- Abdulina, L. et al., "QA in the Age of AI-Accelerated Development"
- Hao He et al., "Speed at the Cost of Quality: How Cursor AI Increases Short-Term Velocity and Long-Term Complexity in Open-Source Projects"
- Google Cloud DORA Report (2025): State of AI-assisted Software Development
- ブロッコリーさんのブログ：20260413 翻訳記事「AIコーディングツールによって加速するコード生成に品質保証活動はどう立ち向かうか」
- Meng & Jackson, ACM Onward! '25., What You See Is What It Does: A Structural Pattern for Legible Software

Appendix

- LayerX Engineer Blog：LLMの「聞きすぎ」を止める：ラベル付きデータで自己分析させたプロンプト改善（2026-05-19）
- ダニエル・カーネマン(著) 村井 章子(訳) 「ファスト&スロー 上&下 —あなたの意思はどのように決まるか？—（ハヤカワ文庫NF）」
- ロバート・B・チャルディーニ 著 社会行動研究会 監訳 「影響力の武器 [新版] 人を動かす七つの原理」
- JSTQB FL V4.0（JSTQB® 2023V4.0.J02）

ユニファ開発チーム、仲間を探しています



[ユニファ株式会社 会社紹介資料（開発チーム版） / Unifa inc
information for dev team - Speaker Deck](#)

ご清聴、ありがとうございました！