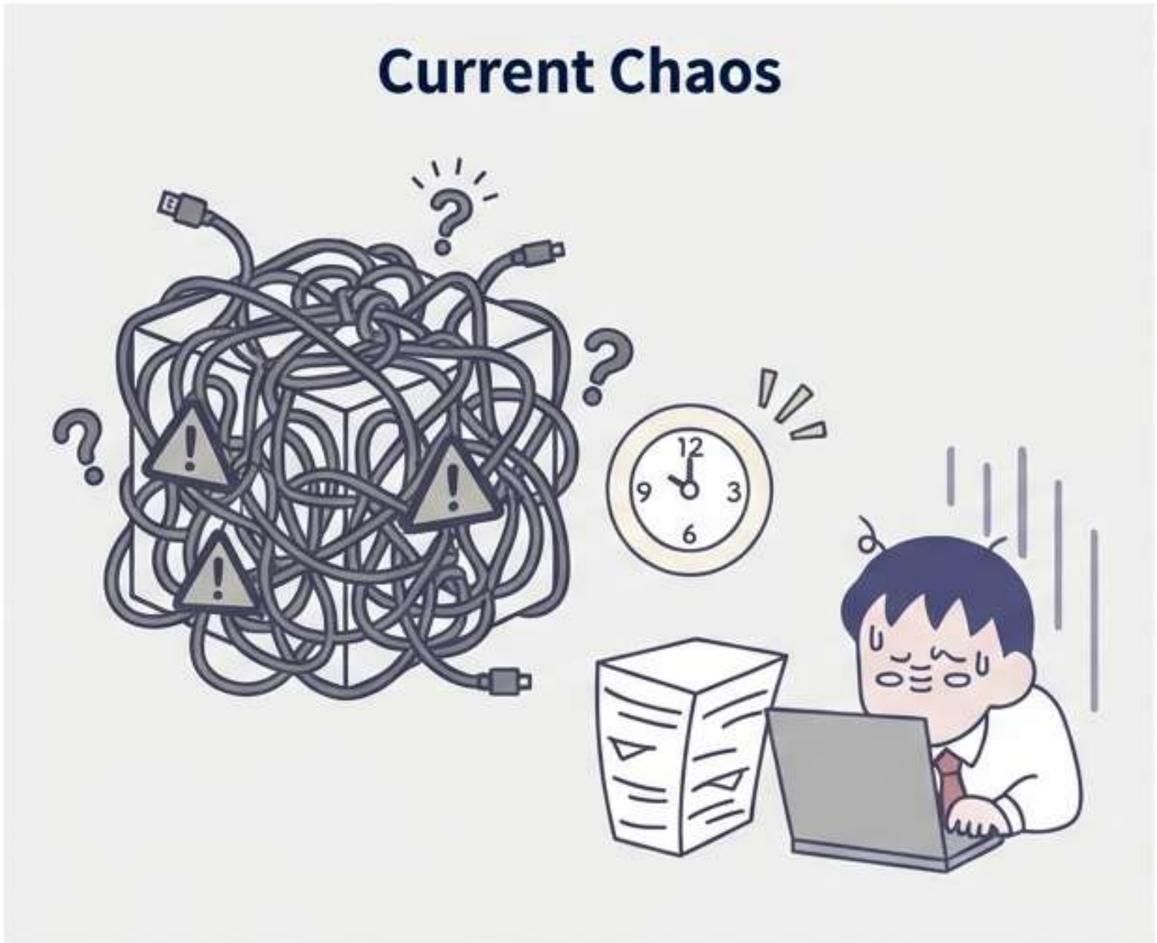
BACKSLASH DESIGN

ソフトウェア開発委託における「構造品質」の標準化とリスク低減のご提案

属人化した「動く秘伝のタレ」から、資産となる「シンプルで堅牢なコード」へ

納品されるコードは「資産（ASSET）」ですか？
それとも「負債（DEBT）」ですか？

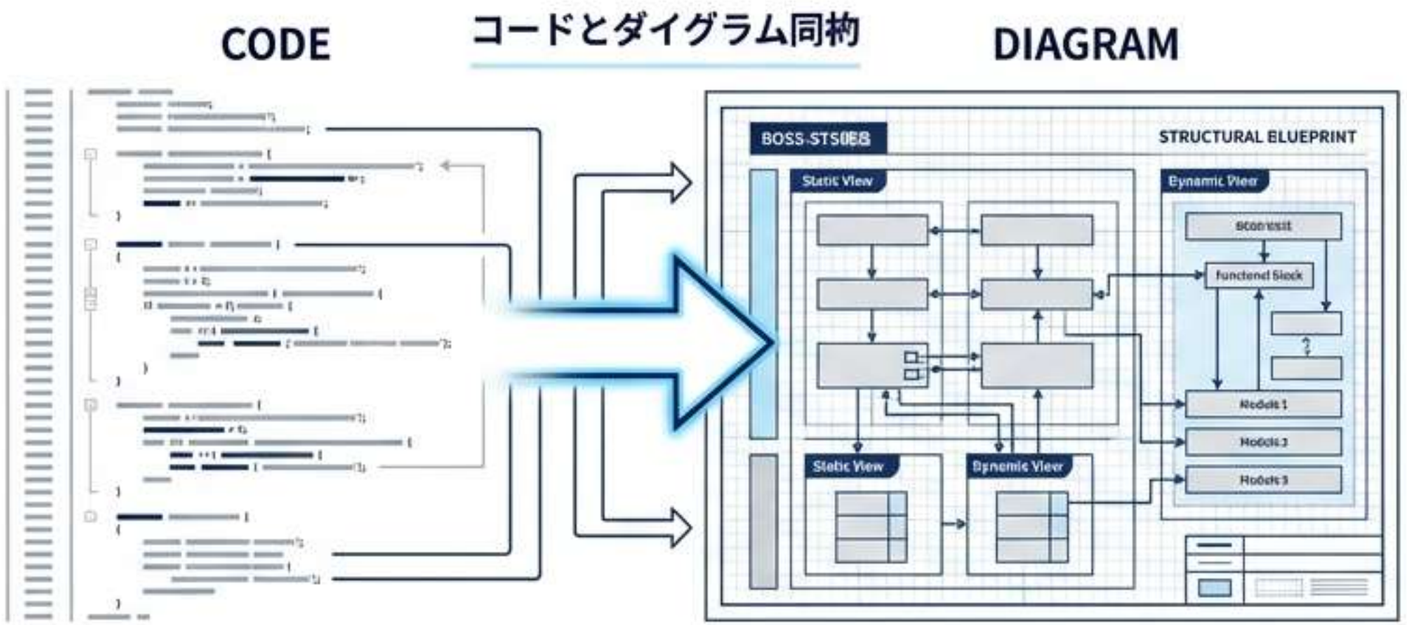


Problem

- **ブラックボックス化**: 仕様書がなく、ソースコードでしか仕様を語れない状態。
- **属人化**: 担当ベンダーが変わるとメンテナンス不能になる「動く秘伝のタレ」。
- **構造劣化**: 機能追加のたびにバグが発生し、テスト工数が肥大化する。

原因：従来のベンダー管理は「機能要件（動くか）」のチェックに留まり、「構造品質（直しやすく壊れにくいか）」の基準が存在しないため。

構造設計の共通言語「AtDesign」による品質の可視化



C言語開発における
「構造設計（アーキテクチャ）」の
スキルを標準化する教育プログラム
「AtDesign」を、
パートナーエンジニアの
スキル要件として推奨します。

共通言語化

ベンダーが異なっても、同じ「地図（BOSS-STS構造）」でコードが記述されるため、可読性が担保される。

品質保証

「なんとなく動く」ではなく、「静的・動的ビュー」に基づいたロジックで実装される。

脱・属人化

「コードでしか説明できない」状態から、「図面で意図を説明できる」状態へ。

「学習」ではありません。「品質特性」の作り込みです。

静的ビュー (Step 1)	• 関数分割 • 高凝集・疎結合 • システム形状（BOSS-STS構造）	**保守性（直しやすさ）／移植性（使い回しやすさ）**

論理構造をシンプルにし、「どこで何をしているか」を明快にする。

動的ビュー (Step 2)	• イベント駆動 • 状態変数 • タスク構造図 • 静動分離	**信頼性（バグのなさ）／効率性（パフォーマンス）**

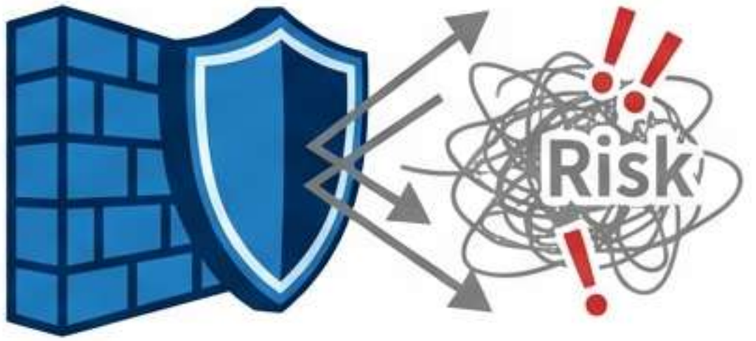
動的な不具合を未然防止し、「どのように動くか」を堅牢にする。

「11,000円」の投資で、受け入れリスクを最小化する

Cost

エンジニア1名あたり
税込 11,000円（Step 1 / 1ヶ月）

Protection



ROI：手戻りやバグ調査にかかる膨大なコストと
比較すれば、圧倒的に安価な「品質保険」です。

Two-Step Action Plan

Step 1: 調達要件への追加

新規参画するエンジニアに対し、AtDesign Step1（静的ビュー）の受講・認定を推奨要件とする。

Step 2: 受け入れ基準の明確化

納品コードが「BOSS-STS構造」等の設計原則に沿っているか確認する運用を取り入れる。

構造を崩さず、設計意図を図面で語れるエンジニアとプロジェクトを成功させるために。

ビースラッシュ株式会社

ホームページ：https://www.bslash.co.jp/
お問い合せ　：info@bslash.co.jp

ホームページ

ソフトウェア
設計力向上セミナー

設計意図発掘ツール
AtScope

ソースコード診断
サービス

アーキテクチャ博物館
AtMuseum

資産レベル判定