

事例に基づくシンプルデザイン

相澤 亮太, 青山 祐三, 小牟田 俊介, 中尾 陽一
 廣田 智紀, 深瀬 智紀, 山本 淳一

背景

近年ウォーターフォール開発（以下WF開発）だけではなくアジャイル開発（以下AG開発）の導入が進んでいるが、AG開発に関する理解度が低く、AG開発に期待する成果が十分に得られていないという課題がある。
 主要な原因は、AG開発の経験値不足（人材不足、プロセス未理解）である。

主旨/ゴール

主旨
 ①AG開発のプラクティスを実践することでメリットを体感する。
 ②WF開発とAG開発の適材適所(メリット/デメリット)を理解する。
 上記より、プロジェクトの性質に応じて最適な開発手法を選択する方法論を提案する

アプローチ/結果/特徴

2チームに別れて、同じ例題をWF型とAG型で開発してその成果物を比較分析する。

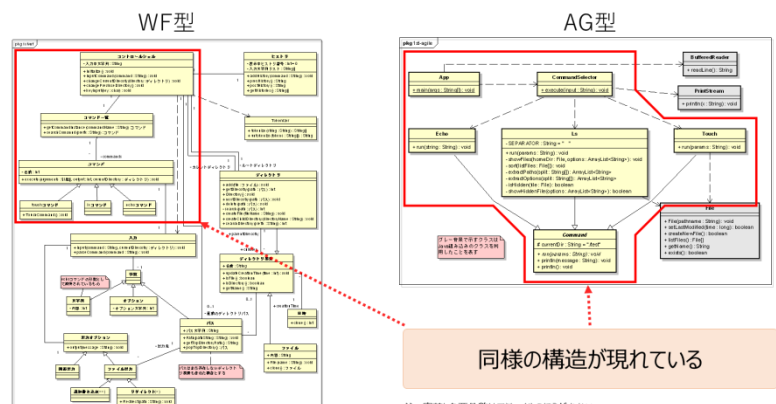


図1: 詳細設計レベルのクラス図比較

演習結果より、各開発手法の特徴に関する様々な見解を得た。

- ・ウォーターフォール開発
 - 全体像が見えているためスケジュールや見積りが立て易い
 - 意見が発散しやすく合意形成が図り難い
- ・アジャイル開発
 - プロジェクトの後期段階のスケジュールが不明確であり、全体的なスケジュールを立て難い
 - 小規模の機能に集中することにより十分な検討ができ、合意形成が図り易い
- ・共通
 - 品質は開発プロセスよりもプラクティスや技術者の熟練度の影響が大きい
 - 開発プロセスに関わらず、ペアワークは有効なプラクティス

考察

以上の演習結果より、開発手法（ウォーターフォール開発/アジャイル開発）は、システム開発の規模/期間に合わせて使い分けることを提案する。

結論/今後の課題

- 以上より、下記結論を得た。
- 開発手法は、システムの開発規模と開発期間を基準に決定する。（大規模な長期開発は要件確度も基準とする）
 - AG開発の経験値を増加させるために、小規模なシステム開発は全てAG開発で実施することが望ましい。
 - AG開発のプラクティスの1つであるペアワークはWF開発でも活用可能。

- 今後の課題として下記3点が挙げられる。
- 実装する要件数を揃えた上での設計比較の実施
 - 成果物の品質を測るためにコード比較の実施
 - 規模や期間の大きい開発での追加検証の実施

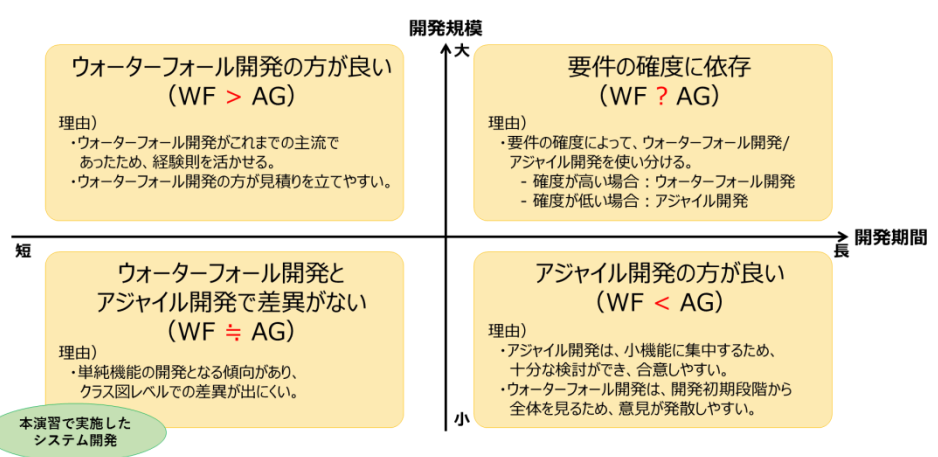


図2: システム開発と開発手法の関係性