

テストプロセス改善を考慮した 少人数によるソフトウェア開発手法の提案

喜多 義弘
長崎県立大学
kita@sun.ac.jp

池田 暁
筑波大学大学院
s2430124@u.tsukuba.ac.jp

岡田 朝日, 森川 慎之介, 本村 亜唯
長崎県立大学
{bs221016, bs221075, bs221074}@sun.ac.jp

馬場 竜一, 水田 真之介, 永尾 直樹, 笹野 真諭
株式会社 NDKCOM
{rbaba, s-mizuta, nagao-naoki, sasano-masato}@ndkcom.co.jp

要旨

ソフトウェア製品の品質を向上させるため、ソフトウェア開発におけるテストプロセスを改善することは有効な手段の1つである。テストプロセス改善では、第三者視点による不具合検知の向上や、製品設計とテスト設計の分担による効率化を図るため、開発チームから独立したテスト組織を編成することが推奨されている。しかし、テスト技術者の人材が不足しているソフトウェアベンダーにとっては、そのことが改善活動の大きな障壁となっている。

そこで本研究では、テスト組織を編成せずともテストプロセス改善を可能とすることを目的とし、少人数による小規模開発手法である「ペア開発」を提案する。ペア開発の実用性を検証するため、実際の開発現場に適用し、その作業時間が従来との差がないことを確認した。

1. はじめに

ソフトウェア開発において、ソフトウェア製品の品質を高めるために、テストプロセスを見直し、改善する動きが高まっている。テストプロセスを改善することは、製品の品質向上につながるだけでなく、テストを効率よくかつ効果的に行うことにより工数とコストの削減にもつながる。

テストプロセス改善では、第三者視点による不具合検知の向上や、製品設計とテスト設計の分担による効率化を図るため、開発チームから独立したテスト組織を編成することが推奨されている [1]。しかし、テスト技術者の人材不足などの理由により独立したテスト組織を編成できない場合があり、そのことが改善活動の大きな障壁となっている。

そこで本研究では、テスト組織を編成せずともテストプロセス改善を可能とすることを目的とし、少人数による小規模開発手法を提案する。

2. 関連研究

2.1. TPI NEXT によるプロセス改善

TPI NEXT[1] とは、ビジネス主導によるテストプロセス成熟度評価とそれに基づく改善を枠組みとしたプロセス改善モデルである。テストプロセス成熟度評価は、テスト活動に必要な要素を 16 のキーエリアと 4 段階の成熟度レベルを定めている。さらに、各キーエリアとその成熟度レベルを確認できるものとしてテスト成熟度マトリクスが設けられている。

図 1 にテスト成熟度マトリクスを示す。成熟度レベルのうち、コントロールレベル、効率化レベル、および最適化レベルにはさらに 3~4 項目のチェックポイントがあり、それらのチェックポイントの満たし方により成熟度を測定する。成熟度を高めるため、TPI NEXT ではキーエリアごとに成熟度レベルに合わせた改善手法が提案されている。

今回、我々はそのキーエリアの中で「テスト組織」に着目した。テスト組織については、「テスト組織とは、独立した存在としての組織」として定義され、テスト関連の成果物やサービスを明確にし、テストに対して専任であり、責任を持つこととしている。しかしながら、人材不足が起きている現場においては、開発チームから独立したテストチームを組織することは難しいことが現状である。

TPI NEXT に関連する既存研究として、TPI NEXT の導入方法 [2] やその支援手法 [3] は提案されているが、人材不足によりテスト組織の編成が困難である場合の対策については提案されていない。

2.2. 従来の開発モデルとその問題点

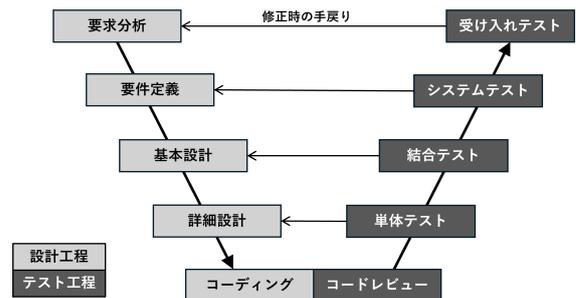
ウォーターフォール型開発モデルをベースとして、設計工程とテスト工程を分離した V 字モデルと、設計工程とテスト工程を並行する W 字モデルがある。図 2 に、V 字モデルと W 字モデルの流れを示す。

V 字モデルは、設計工程とテスト工程が分離しているため、それぞれの役割に分担しやすいメリットを持つ。一方、W 字モデルは、V 字モデルのメリットを併せ持ち、さらに製品の設計をしながらテスト設計を行うため、

キーエリア	成熟度レベル												
	初期レベル	コントロールレベル				効率化レベル				最適化レベル			
1 利害関係者のコミットメント		1	2	3	4	1	2	3	4	1	2	3	4
2 関与の度合い		1	2	3	4	1	2	3	4	1	2	3	4
3 テスト戦略		1	2	3	4	1	2	3	4	1	2	3	4
4 テスト組織		1	2	3	4	1	2	3	4	1	2	3	4
5 コミュニケーション		1	2	3	4	1	2	3	4	1	2	3	4
6 報告		1	2	3	4	1	2	3	4	1	2	3	4
7 テストプロセス管理		1	2	3	4	1	2	3	4	1	2	3	4
8 見積もりと計算		1	2	3	4	1	2	3	4	1	2	3	4
9 メトリクス		1	2	3	4	1	2	3	4	1	2	3	4
10 欠陥管理		1	2	3	4	1	2	3	4	1	2	3	4
11 テストウェア管理		1	2	3	4	1	2	3	4	1	2	3	4
12 手法の実践		1	2	3	4	1	2	3	4	1	2	3	4
13 テスト担当者のプロ意識		1	2	3	4	1	2	3	4	1	2	3	4
14 テストケース設計		1	2	3	4	1	2	3	4	1	2	3	4
15 テストツール		1	2	3	4	1	2	3	4	1	2	3	4
16 テスト環境		1	2	3	4	1	2	3	4	1	2	3	4

図 1: テスト成熟度マトリクス [1]

V字モデル



W字モデル

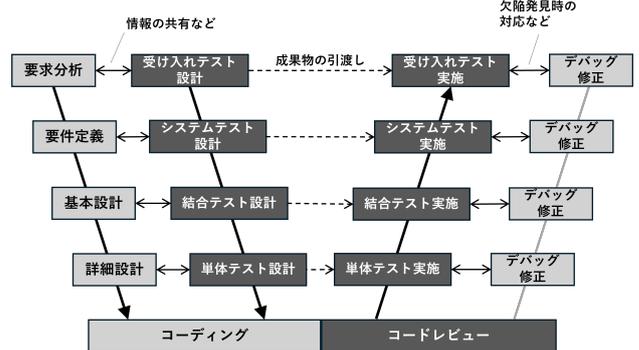


図 2: V 字モデルと W 字モデルの流れ

テストの観点を取り入れやすく、製品設計の漏れや誤りを早期に発見しやすいメリットを持つ。

V字モデルの問題点は、テスト工程が進むほど当工程で発見した欠陥の修正による手戻りは大きく、コストの増加や工期の圧迫が発生する点である。一方、W字モデルはこのV字モデルの問題点を設計工程とテスト工程の並行により改善しているが、W字モデルの問題点として、初期のテスト設計が難しい点が挙げられる。これは初期の設計工程において、開発する製品の要求や要件を整理する前にテスト設計を行うことが起因している。

2.3. アジャイル開発導入の課題

開発途中における要求仕様の変更に対応するために、アジャイル開発を導入することが増えている。アジャイル開発 [4] とは、漸次的に進めていく開発を反復的に繰り返していく開発手法である。この手法では、作成したアウトプットに基づいて次に取り掛かる開発の最適化を行うため、開発途中での変更に対応しやすい。

しかしながら、アジャイル開発は従来の開発モデルとは異なるため、その導入には以下の課題を抱えている。

- 開発に関わる全ての者がアジャイル開発について理解しておく必要がある
- アジャイル開発に特化したチームを編成する必要がある
- チーム内のコミュニケーションを密に取り合うことが必要である

特に人材不足の問題が深刻であるソフトウェアベンダーでは、アジャイル開発に特化したチームを編成することは難しく、組織内でのアジャイル開発に向けた人材育成や技術研鑽の場を持ちにくい。

3. 提案

本研究では、テスト組織を編成せずともテストプロセス改善を可能とすることを目的として、少人数による小規模開発手法を提案する。具体的には、最少人数を2人とし、それぞれを開発の責任者（以降、製品設計者とする）とテストの責任者（以降、テスト設計者とする）

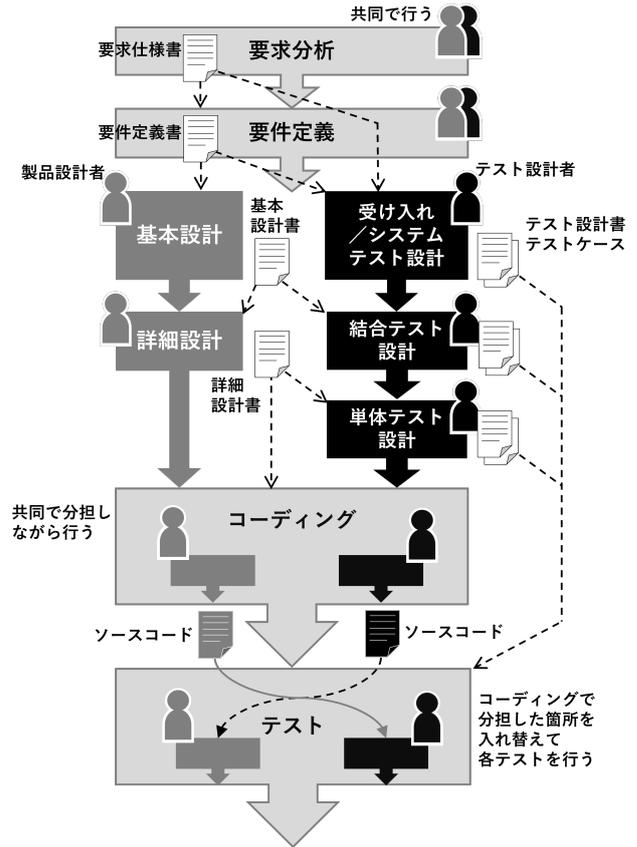


図 3: ペア開発の流れ

にすることで、それぞれの活動や責任所在を明確にする手法である。本論文において、提案する開発手法を「ペア開発」と称する。

なお、このペア開発において、重視する点は以下の3点である。

- テスト組織（テスト設計者）の独立性
- テスト駆動開発の容易性
- メンバー間のコミュニケーション

これらの点を踏まえながら、以降ではペア開発の全体像と各役割の責任について述べていく。

3.1. ペア開発の全体像

ペア開発は、ウォーターフォール型開発モデルを考え方ベースとする。この考え方を導入する理由として、従来親しまれていることから開発全体の流れを理解しやすく、それぞれの役割を明確にしやすいためである。ま

た、人材の確保や開発規模の拡大が進み、ペア開発から派生または移行することを考慮し、基本的な開発手法として位置づける。

図3に、ペア開発の流れを示す。この流れは最少人数である2人で行うことを想定した流れである。

まず要求分析および要件定義を、2人共同で行い、仕様書を共同で作成する。その理由として、以下の2点を挙げる。

- 要求を漏れなく引き出すため

互いの観点を尊重しつつ、共に顧客と接することにより、顧客から要求を引き出しやすくする。また、引き出し役と聞き役に役割を分けることにより、聞き役は引き出し役の引き出し方に対して客観的に俯瞰できるため、漏れや誤解を防ぐことができる。

- 要求および要件を相互に認識し、設計時における要件の誤解や認識のずれを防ぐため

要求および要件を深く理解でき、それを相互に確認し合うことにより、設計時において要件の誤解や互いの認識のずれを防ぐことができる。また、製品に対する個々の理解度も上がり、質の高い設計ができることが見込まれる。

次に各役割において、基本設計とテスト設計に分かれ行う。この際、互いに干渉することなく、独立して作業を行う。その理由として、本章の冒頭で述べた重視する点「テスト組織（テスト設計者）の独立性」を確保するためである。

基本設計と同時にテスト設計を行うことにより、コーディング実施前の基本設計書や詳細設計書と同時期に、テスト設計書やテストケースが出来上がる。これにより、テスト駆動開発 [5] の基本概念を導入することが容易になる。テスト駆動開発の考え方は、従来のウォーターフォール開発モデルでは馴染みがないため、図3の全体像においてはコーディング時にテスト設計書やテストケースを用いていない。しかしながら、アジャイル開発への移行を考えている現場においては、ペア開発はテスト駆動開発を導入しやすい形であり、移行の初期段階で採用しやすいと考えられる。

次にコーディングとなり、基本設計書および詳細設計書に従い、作成箇所をそれぞれで分担しながら各自で行

う。分担の方針は任意であり、製品設計者が中心となって分担する、各自の得意な箇所や技量に合わせて分担するなど、作成する規模や期間に合わせて柔軟に決定する。

最後はテストになる。ここでは、コーディング時に担当した箇所以外をテストする。テストケースは事前に作成しているため、どちらが担当してもテスト結果は変わらないが、テスト結果に対してもより客観的に捉えることができ、欠陥の検出率向上に寄与できると考える。

また、結合テスト以降のテストにおける、それぞれが担当した箇所を合わせて同時にテストを行う場合には、双方ともに共同でテストを行う。その際に検出された欠陥については、当該箇所のコーディング担当以外の者が欠陥の記録などを行う。

3.2. 各役割の責任

テストの専任者を立てることで、テストの管理を一元化し、テスト一貫の作業に対して責任を持ち、専念させることが望ましい。しかしながら、人材不足である場合はテストの専任者を立てることは難しい。そこで2人体制の中で責任の所在を明確にすることにより、役割への意識を高めつつ、負担を分散することを目指す。

ペア開発における各作業の責任は以下のとおりである。

- 要求分析：製品設計者およびテスト設計者
- 要件定義：製品設計者およびテスト設計者
- 基本設計：製品設計者
- 詳細設計：製品設計者
- テスト設計：テスト設計者
- テストケース作成：テスト設計者
- コーディング：製品設計者
- テスト：テスト設計者

要求分析および要件定義の責任者を両者にした理由として、作業に対する意識を高め、要求や要件を十分に理解し、作成する仕様書について責任を持つためである。

各設計については、担当した箇所について責任を持つこととし、コーディングやテストについては、そこで利用する設計書やテストケースに付随する形で責任を持

たせている。また、テスト設計者がテストに対して専念できるよう、テスト関連の作業とそれ以外の作業で分けた形である。

4. 評価

ペア開発の実用性を確認するために、ソフトウェア開発の現場にて試用実験を行った。当現場では従来、製品設計における詳細設計終了後に、テスト設計を行っている。製品設計者とテスト設計者は別担当であるが、テスト設計者は詳細設計書などすべての設計書がある状態でテスト設計を行うことが可能である。また、要求仕様書や要件定義書は製品設計者が作成を担当しており、テスト担当者は製品設計後に作業へ関わる。

図4に、実験の概観を示す。今回の実験では、既存プロジェクトにおいて作成済みの要求仕様書および要件定義書を用いて、基本設計、詳細設計、およびテスト設計を行い、各成果物を作成するのにかかる時間を計測する。また、既存プロジェクト内の成果物と当実験での成果物について内容の比較も行う。当実験の被験者として、既存プロジェクトには関わっていない技術者2名に、製品設計者またはテスト設計者として分担する。

被験者である各技術者は、JSTQB 認定テスト技術者資格 [6] を有しており、各担当において適任であると判断した。また被験者は、各作業を始める前に要求仕様書および要件定義書を読解し、被験者同士が互いに内容を理解し合うまで打合せを行った。その時間も「事前準備」として別途計測する。各作業を始めた後も、作業の分担は厳密に守りながらも、必要であれば互いに随時コミュニケーションを取ることにした。

表1に、各作業の所要時間を示す。合計時間は各工程の時間の和を示す。想定時間は、ペア開発において「基本設計/詳細設計」と「テスト設計」は並行であると想定し、実験における当工程の時間が短い方（表中ではテスト設計）を合計から差し引いた時間である。

表1より、ペア開発は既存プロジェクトと比べて、合計時間は時間が長いですが、想定時間を考慮した場合は0.5時間短い結果であった。

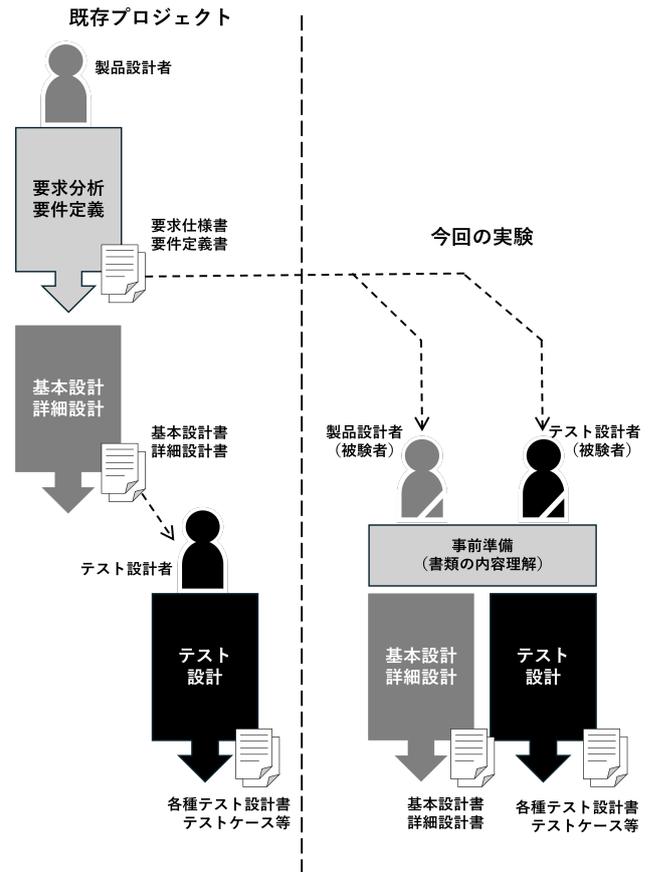


図4: 実験の概観

表1: 各作業の所要時間 (時間)

	既存プロジェクト	ペア開発(実験)
要求分析/要件定義	10.5	-
事前準備	-	8.5
基本設計/詳細設計	4.5	19.5
テスト設計	13.5	5.0
合計時間	28.5	33.0
想定時間	-	28.0

このことから、ペア開発は実用的な時間で実施することが可能であると考えられる。そして、各工程にかかった時間として、「基本設計/詳細設計」は既存プロジェクトより15時間長く、「テスト設計」は既存プロジェクトより8.5時間短い結果であった。

5. 考察

5.1. 実験結果の考察

表1において、ペア開発の「基本設計/詳細設計」が既存プロジェクトよりも15時間多い理由として、設計の粒度に違いが出たためと考えられる。製品設計者を担当した被験者によると、並行で作業を行うテスト設計者を意識し、記述した内容がコード寄りの詳細ものが多くなった、と感想を述べていた。また、事前準備の段階で、テスト設計者との認識合わせを十分に行ったことは、テスト設計者を意識することにつながる、との見解も示しており、製品設計の観点とテスト設計の観点から互いに意見を出し合うことができていた。互いの意見が異なる場合は、分析や認識が足りない可能性を考慮し、より重点的にチェックできた、との見解も示した。

一方、ペア開発の「テスト設計」が既存プロジェクトよりも8.5時間短い理由として、「基本設計/詳細設計」と同様に設計の粒度に違いが出たためと考えられる。設計書の内容が、既存プロジェクトよりも粒度が粗く、洗い出した項目も少なかった。これは基本設計書や詳細設計書を反映していないため、抽象的な内容に留まったと考えられる。

今回の実験では試みていないが、「基本設計/詳細設計」で得た成果物を用いて「テスト設計」を行うことにより、粒度をより細かくできると考えられる。「基本設計/詳細設計」以降の工程については、今後に追実験を行い、改めて検証する必要がある。

5.2. 従来モデルとの比較や課題解決に対する考察

従来モデルであるW字モデルの課題である、初期のテスト設計が難しい点について、ペア開発では、要求分析および要件定義の工程を製品設計者およびテスト設計者の両名で行うことにより、製品に対する各要求や要件を整理した上でテスト設計を行うことでその課題を解決できると考える。また、テスト設計者もそれらの工程に介入することで、テスト設計の観点を持って要求分析や要件定義を行うことができる。これはW字モデルと同様のメリットであり、これらのことからペア開発はW字モデルの上位的な開発手法であると言える。

しかしながら、W字モデルでは要求分析と同時期に

テスト設計を開始しているのに対し、ペア開発は要件定義終了後にテスト設計を開始している。このことから、W字モデルよりも工期が延びることが示唆される。

また、W字モデルでは、テスト設計をすることでその成果物が具体的になるため、その成果物を作成しながらの要求分析ではより具体的に要求仕様書の漏れや誤りを指摘しやすい。ペア開発では、テスト設計は要件定義後にテスト設計を開始するため、その成果物がない状態での要求分析や要件定義になる。そのため、テスト設計に長けている者がテスト設計者であることが望ましく、そのことはW字モデル導入時よりも求められると考えられる。

5.3. アジャイル開発モデルへの派生

ペア開発は、コーディングの前にテスト設計を行うため、テスト駆動開発[5]の考え方をを用いることができる。また、コーディングの部分をペアプログラミングに置き換えることもできる。これらにより、ペア開発はアジャイル開発との親和性が高いと考えられるが、ペア開発のベースがウォーターフォール開発モデルであるため、アジャイル開発への移行は従来と同等の時間とコストがかかる見込みである。

ペア開発では、要求分析や要件定義において各設計者の認識を十分に取ることに重点を置くため、アジャイル開発モデルに近い形への派生を考慮した場合も、この点を重視する必要がある。各設計者の密なコミュニケーションが実現可能である場合は、要求分析や要件定義の工程をより軽量に改良できる余地があると考えられる。

5.4. 今後の課題と展望

ペア開発により、製品設計とテスト設計を分離することで、開発から独立したテスト組織を編成せずとも疑似的にその形を実現することを目指した。しかしながら、このペア開発によってテストプロセス改善が可能であるかを未だ検証できていない。そのため、ペア開発を導入した上でTPI NEXTのテストプロセス成熟度評価[1]やその改善活動を行い、ペア開発の有用性について検証する必要がある。

さらにペア開発は、初級または中級の技術者とベテランの技術者を組み合わせることにより、技術者の育成

のためのフレームワークとして利用できると考える。その場合、ベテラン技術者を製品設計者とテスト設計者のどちらの役割に置くかの議論や、その育成効果についての検証を今後行う必要がある。

6. おわりに

本研究では、テスト組織を編成せずともテストプロセス改善を可能とすることを目的とし、少人数による小規模開発手法であるペア開発を提案した。ペア開発の実用性を確認するため、既存プロジェクトの成果物を利用して、基本設計、詳細設計、およびテスト設計の工程をペア開発によって検証し、実用的な時間で実施できる結果を得ることができた。ペア開発により、少人数での開発規模でテスト組織を疑似的に実現することで、有効なテストプロセス改善を実施できると見込まれる。

参考文献

- [1] A. V. Ewijk, et al., “TPI NEXT® Business Driven Test Process Improvement,” Sogeti Nederland B.V. 2013, 訳 藪田和夫, 湯本剛, 皆川義孝, “TPI NEXT® ビジネス主導のテストプロセス改善,” 株式会社トリフォリオ, 2015.
- [2] 高野愛美, 河野哲也, “品質保証部門におけるテストプロセス改善モデル初期導入に関する取り組み,” ソフトウェア品質シンポジウム 2016, 2016.
- [3] 河野哲也, 山崎崇, 佐藤徳尚, “TPI NEXT による現場主導のテストプロセス改善を支援するための手法の提案,” ソフトウェアシンポジウム 2017, 2017.
- [4] 内閣官房情報通信技術 (IT) 総合戦略室, “アジャイル開発実践ガイドブック,” 2021.
- [5] K. Beck, “Test Driven Development by Example,” Pearson Education, Inc., 2003, 訳 和田卓人, “テスト駆動開発,” 株式会社オーム社, 2017.
- [6] JSTQB 認定テスト技術者資格, <https://jstqb.jp/>