

2つの立場で訓練する反復型トレーニングによる技術者育成

- XDDP 成果物の質向上を目指したトレーニング手法の提案 -

Development of engineers by repeating training on the two positions

- Approach a training method to improvement of quality of XDDP deliverables -

デンソーテクノ株式会社

DENSO TECHNO CORPORATION

○中井 栄次	飯尾 将実	内田 有紀	山路 厚
○Eiji Nakai	Masami Iio	Yuki Uchida	Atsushi Yamaji

Abstract In this paper, we describe the proposal of training method improved engineers technical skill and quality of deliverables. We also report the result of our activity applied this method for improvement of XDDP. The feature of this training method is to experience both “be instructed” and “instruct” at the same time and to introduce the perspective of an outsider. Concretely, we combined the instruction from external consultant and the instruction to members by using design review. These approaches improved the skill of engineers and the quality of XDDP deliverables.

1. はじめに

我々はカーナビゲーション（カーナビ）のソフトウェアを開発している。近年、カーナビのソフトは、高機能化・低コスト化・短納期化が急速に進んでおり、従来の開発プロセスではプロジェクトのQCD目標を達成することが困難になっていた。そのような状況において、我々はカーナビのソフト開発のほとんどが派生開発であることに着目し、派生開発に特化した開発プロセスXDDPを2010年頃から導入した。XDDPでは、ベースソフトからの変更箇所を変更要求仕様書に全て書き出し、変更箇所に着目したレビューを実施する。そのため、変更箇所の抜け漏れを未然に防止し、QCDを向上することが期待できる。

しかし、実際には、上流工程で変更箇所の抜け漏れを十分抑えることができず、後工程に不具合が流出し、手戻りが発生していた。その要因を分析した結果、作成した変更要求仕様書の構成や表現に問題があることがわかった。具体的には、変更要求仕様書の階層構造やグループが整理されておらず抜け漏れに気づきにくい、あるいは変更前の状態を正確に表現できていないため、変更に対する影響範囲に気づきにくい、といった問題があった。これらの問題は、担当者がXDDPのねらいや効果を理解しないまま変更要求仕様書のフォーマットを埋めるという形式的な“作業”を行っているために発生していた。我々は、技術者育成の観点から、XDDPの教育で獲得する知識が実践で活用できる形に整理されておらず、かつ教育・業務において、担当者が成果物の質を向上することに対して受け身の立場にいることが根本的な要因であると考えた。

そこで、我々は、XDDPのねらいや効果を理解し、高品質な成果物を作成できる技術者を育成する活動を実施した。具体的には、実践的な教育とレビューを用いた実践指導を組み合わせ、「教えられる・教える」立場を同時期に経験するトレーニング手法である。本トレーニングを若手リーダーに適用し、対象者のXDDPスキル・現場の成果物の質・品質状況の3点で評価を行い、一定の効果があることを確認できた。

2. 現状と課題

2.1 XDDP とその成果物

XDDP は、清水吉男氏が提案した派生開発専用の開発プロセス^[1]である。派生開発とは、過去に開発した製品に機能追加や性能の改良などを行い、新しい製品として市場に出す開発である。XDDP のプロセスを図 1 に示す。XDDP の最大の特徴は、機能追加と変更のプロセス・成果物を分け、ベースソフトからの変更箇所に着目して開発を進めることである。変更のプロセスでは、派生元のソースコードを変更する前に、「何を」「どこで」「どのように」変更するかという観点でそれぞれ変更要求仕様書、TM（トレーサビリティマトリクス）、変更設計書の 3 つの成果物を作成する。変更要求仕様書には、実装仕様レベルの変更箇所を全て書き出し、関係者とのレビューによって、変更箇所の抜け漏れ・誤りを防止する。

XDDP における変更要求仕様書は、USDM(Unified Specification Description Language)^[2]により記載する。図 2 に USDM の構造を示す。要求は機能・性能・制約条件を抽象的に表現した一連の振る舞いとして表現する。そして、要求から具体的な仕様を引き出し、要求と同じ枠内の一階層に記述する。要求・仕様の数が増える場合は、グループでカテゴリ化することもできる。このように USDM では、要求と仕様を階層的に表現することにより、仕様の抜け漏れを防止している。これは、担当者による抜け漏れの防止だけでなく、レビューで関係者が抜け漏れに気づきやすくなることもねらっている。

派生開発では、変更による影響範囲の認識不十分・検討不足から変更仕様が増えることが多い。変更による影響範囲の抜け漏れを抑えるためには、ベースソフトの仕様を段階的に詳細化して影響範囲を検討することが効果的である。変更要求仕様書は、USDM を用いて変更箇所を表現することで、変更に対する影響範囲を特定しやすくしている。

しかし、USDM のフォーマットは自由度が高く、その効果は作成する技術者のスキルに依存しやすい。例えば、図 2 において、上位の要求(Req. 1)の直下に全ての仕様(Req. 1-1-1~Req. 1-2-4)を書き出すことも可能だが、可読性が悪化するため仕様の抜け漏れに気づきにくくなる。このように、抜け漏れを防ぐための階層構造や変更内容が適切に表現されなければ、USDM で記述しても効果を得ることができない。つまり、XDDP を用いた開発を成功させるためには、いかに変更要求仕様書に抜け漏れなく変更箇所を記載し、レビューが抜け漏れ・誤りを指摘できるような表現で記載するかが重要なポイントとなる。

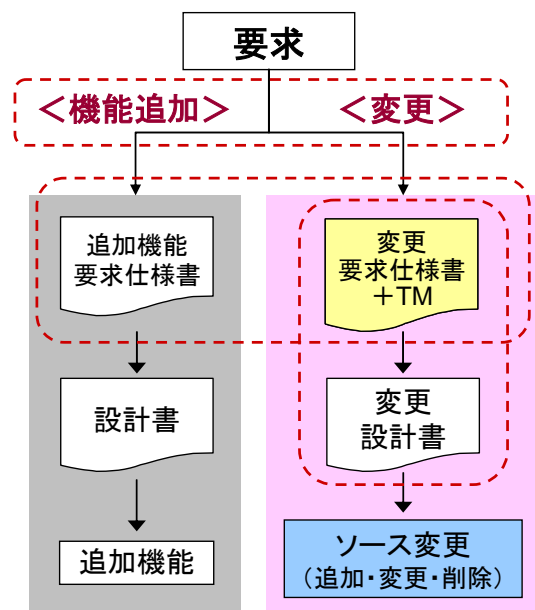


図 1. XDDP

要求	Req.1	
	理由	
	説明	
<Group 1>		
要求	Req.1-1	
	理由	
	説明	
仕様	<グループ A>	
	Req.1-1-1	
	Req.1-1-2	
	<グループ B>	
	Req.1-1-3	
	Req.1-1-4	
<Group 2>		
要求	Req.1-2	
	理由	
	説明	
仕様	<グループ C>	
	Req.1-2-1	
	Req.1-2-2	
	<グループ D>	
	Req.1-2-3	
	Req.1-2-4	

図 2. USDM

2.2 XDDP 適用の現状

我々はカーナビ開発の大半を派生開発で行ってきた。そのなかで、2010年頃からXDDPを導入し、ソフトウェアの品質向上を図ってきた。その結果、工程内の不具合による手戻りが減少し、生産性の向上が確認できた。これは、XDDPで定義された成果物を作成することにより、従来は担当者に依存していた変更箇所の抽出プロセスを一定のレベルに引き上げることができたためである。しかし、残存する不具合の要因を調査したところ、大半の要因が仕様の検討漏れで発生していた。これは、我々がXDDPで本来解決したい課題を十分解決できていないことを表していた。

2.3 課題

上述した品質の状況に対して、我々はXDDPの成果物である変更要求仕様書の質に問題があると考え、調査を行った。実際の開発で作成された変更要求仕様書を査読し、階層構造や変更の表現を分析したところ、表1のような問題の傾向があることがわかった。

表1. 変更要求仕様書の代表的な問題

問題	問題の詳細
無意味な(過剰な)階層化	USDMの可読性が悪化する。変更要求や変更仕様の抜け漏れに気づきにくい。
網羅性のないグルーピング	グルーピングの観点が統一されておらず、抜け漏れに気づきにくい。
テンプレート化された階層表現	案件の特性によらず、階層構造やグループ分割がテンプレート化されている。
あいまいな変更表現	結局何が変更されるのかわからないため、影響範囲に気づきにくい。

これらの問題は、変更要求仕様書をどう記述すれば変更箇所の抜け漏れを防止できるかを担当者が理解していないことを表している。さらに、担当者へのヒアリングを行った結果、担当者は与えられたプロセス・成果物のフォーマットを形式的に“使っているだけ”であり、XDDPのねらいや効果を十分理解していないことがわかった。

このように、導入・整備したプロセスに対して、開発を担当する技術者がそのねらいや効果を理解しないため期待する効果が得られないという問題は、XDDPに限った問題ではなく、開発現場でよく見かける典型的な問題である^[3]。特に、XDDPにおいては、USDMフォーマットの自由度の高さ故に、技術者には高品質な変更要求仕様書を記述できるスキルが求められる。そこで、我々は、XDDPのねらいや効果を理解し、高品質なXDDPの成果物を作成できる技術者の育成とそのトレーニング手法の確立を課題として設定した。

3. 課題の分析

3.1 着眼点

我々は、従来から座学によるXDDPの教育を実施していた。それにも関わらず、教育で身につけた知識が業務で実践できない点について、教育内容の強化だけで解決することは難しいと考えた。さらに、我々の開発現場では派生開発が五月雨式に発生しており、教育だけに頼らず、業務の中で実践力を磨く工夫が必要であった。そこで、知識と実践の関係・教育と業務における担当者の立場に着目し、以下2つの問題があると考えた。図3に問題の概念モデルを示す。

3.2 知識と実践のギャップによって生じる問題

まず第1は、教育の内容と実践とのギャップによって生じる問題である。従来の教育は、XDDPの汎用的な基礎知識を伝えることに大半の時間を費やし、演習も一般的な内容に対して成果物を作成する、といった内容になっていた。しかし、実業務では、開発の背景や制約により、教育で学んだ知識をそのまま使用できないことが多い。このため、担当者の中で実践的な知識として整理されていない状況では、納期・コストといった実践上の制約を満たすために表面的な知識を用

いて形式的な成果物を作成せざるを得ない。この結果、教育で得た知識が徐々に失われていくことになる。

このように、教育では知識中心、業務では実践中心となっており、互いに補完し合える状況になっていなかった。知識が整理されていないため実践に使用できず、実践の結果が整理した知識にフィードバックされないため、教育の効果が失われる。これは、知識と実践をつなげる教育が不足しているためである。

3.3 立場によって生じる問題

第2に、教育や現場の指導に対して受講者が受け身となり、言われたことを反映するという対応に留まっていたという問題である。教育では受講者がある程度受け身となることは避けられない。そこから実業務に活かすためには、業務の中で考え方を変える、いわゆる行動変容が求められる。しかし、上司や直属のリーダーから強く指導されることにより、「やらなくてはならない」との使命感のもと、「言われた通りに直す」だけとなりやすく、行動変容する機会がなかった。例えば、レビューでの指摘に対して、指摘の背景にある本質的な要因を考えず、指摘された内容だけを修正するといったことが起きる。

また、上司も自分が理解していることに対して、それが常識であると錯覚してしまうことがある。そのため、部下への説明が不親切となり、目的やその状況に応じた効果を十分に説明せずに指導することになる。その結果、部下は目的が理解できず、「とりあえず直す」だけとなりやすくなる。これらは、上司・部下の関係、つまり権限関係が存在するが故に発生する問題である。

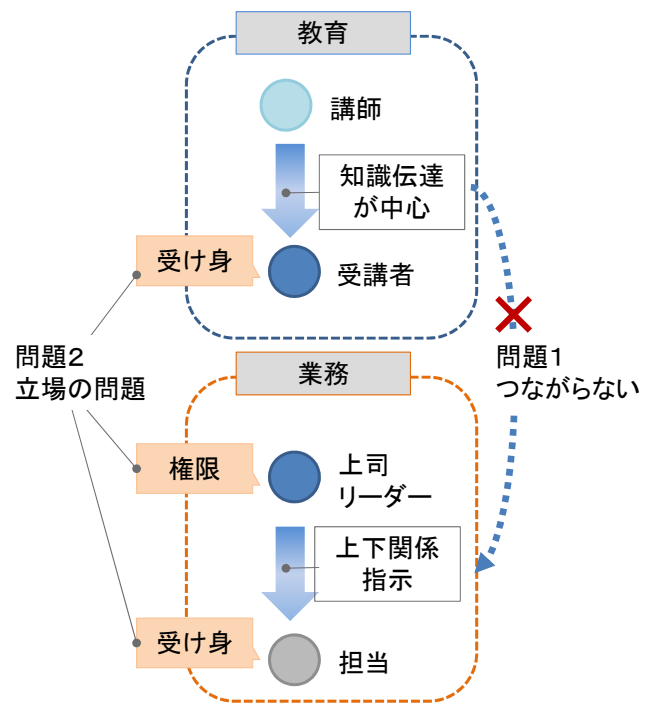


図3. 教育内容と立場によって生じる問題

4. 改善策

4.1 「教えられる・教える」を同時期に経験する手法の提案

我々は「知識と成果物をつなげる」「受け身」という問題を解決するため、「教えられる」「教える」立場を同じ時期に経験する方式を考案した。これは、教えられた知識を実践で活用できるように整理し、それを教えることで知識と実践をつなげ、教える立場で受け身姿勢を払拭させることをねらったものである。さらに、「教えられる・教える」関係に第3者視点を加えることで、権限による強制力を排除し、目的・理論ベースで指導可能となるやり方とした。

さらに、各対策を同時に反復して実施することにより、短期間で実践力を向上することができる考えた（繰り返し行う訓練である）。特に、教えられた内容をすぐに現場で教えることにより理解が深まることを期待した方式である。図4に本手法の概念モデルを示す。以下、それぞれの対策の詳細を述べる。

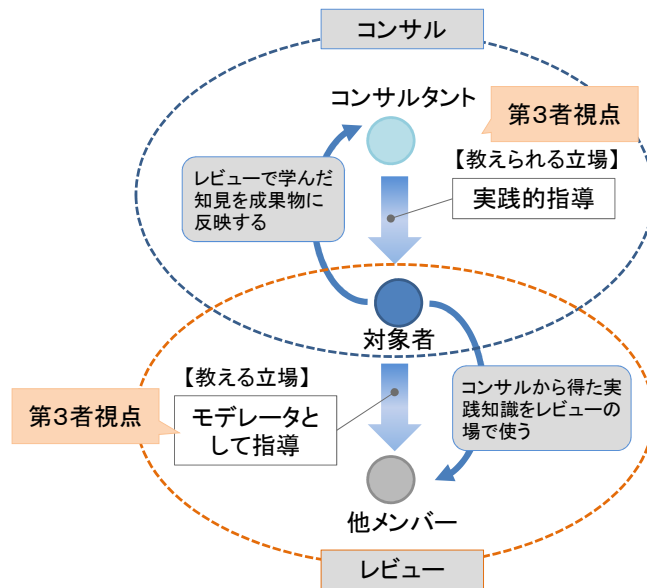


図4. 「教えられる・教える」を同時期に経験する手法

4.2 知識の整理と活用による教育効果の向上（教えられる立場）

まず、変更要求仕様書の質を整理し実践で活用できる知識として定義するため、変更要求仕様書のチェックシートを作成した。具体的には、表2の分類で変更要求仕様書の評価観点を定義し、定義した観点を元に詳細なチェック項目を作成した。また、それぞれのチェック項目にはその理由を記載し、表面的な知識ベースのチェックとならないようにした。

表2. 変更要求仕様書の評価観点

分類	観点
理解性	レビューの効果を高める内容になっているか
妥当性	記述の内容は確からしいか
網羅性	カテゴリやグループの網羅性は確からしいか
効率性	必要以上に記載していないか

次に、社内の教育だけでなく社外のコンサルタントを活用することとした。具体的には、担当者が実開発の成果物を作成し、コンサルタントに指摘・指導してもらう場を定期的に設定した。コンサルの場では、実開発の開発規模・スケジュール・難易度など、開発の背景を踏まえた指摘・指導を受けた。コンサルで受けた指摘・指導に対しては、変更要求仕様書のチェックシートの観点をを用いて指摘の内容を分類した。これにより、担当者は自身にどのような観点を記述が不足しているかを把握することができる。合わせて、担当者は、指摘に対する自分の理解や今後の改善内容について、指摘事項レポートとして残すこととした。これにより、座学では得られない実践的な知見を習得でき、また、知識と実際の成果物とのつながりを理解できる。

さらに、コンサルタントとの間には組織上の上下関係・業務のしがらみがないため、指摘・指導を素直に聞くことができるという第3者視点の効果も期待できる。

4.3 開発現場で指導することによる実践力の向上（教える立場）

次に、担当者が教える立場になることにより、受け身の立場から脱却することを目指した。具体的には、担当モジュール以外のレビューにモデレータとして参加し、レビューを進行させることとした。レビューを対策の場として選定した理由は次の2点である。第1に、レビューは日々繰り返し行われる活動であり、実践的な反復訓練によって短期間でのレベルアップが期待できる。第2に、XDDPにおいては、レビューが有効に機能するかどうか仕様の抜け漏れを防ぐ重要なボ

イントであり、実践においてレビューが最も品質を高める効果があると考えた。

モデレータの責務は、レビューの進行と合わせて、XDDP のねらいや効果を説明しつつ、第3者視点で成果物の質が向上するよう指摘することである。例えば、USDM では要求・グループの階層ごとに抜け漏れがないかを確認していくリーディング方法が効果的である。そのような観点でレビューを進行することで、どのような階層構造にすれば仕様の抜け漏れを防止できるかという知見を蓄積することができる。

あえて担当モジュール以外のレビューに参加する理由は次の2点である。まず、担当モジュールのレビューでは、どうしても機能的に不具合がないかという観点で成果物を査読してしまい、成果物の質という観点がおろそかになってしまうという点である。次に、担当外のレビューでは、上下関係による押し付けや理想論が通用せず、モデレータには指摘するための論理的な説明が必要となる点である。つまり、第3者視点で教える立場に身を置くことで、XDDP のねらい・効果を主体的に考える立場に自然に導く方法である。

5. 実施結果と考察

5.1 実施内容と評価方法

我々は、若手リーダーを2名選出し、本トレーニング手法を試行した。実施期間は、2013年7月から2014年4月までの10ヶ月間である。教育の評価は、該当期間のプロジェクトの特徴や環境の変化といった他要因の影響を受けるため、一面的なデータだけで判断することが難しい。そのため、多面的なデータ分析を実施する必要があると考え、教育の評価方法として一般的なカークパトリックの4段階評価モデル^[4]を適用した。本モデルは、教育の効果を4段階のレベル(Level1:Reaction, Level2:Learning, Level3:Behavior, Level4:Results)ごとに測定する。今回は、Level2からLevel4をそれぞれ以下3つの方法で評価した。Level1は評価のレベルが低く、アンケートによる教育の満足度調査を行うことが一般的であるため、今回は対象外とした。

5.2 コンサルにおける指摘内容の推移 (Level2:Learning)

まず、トレーニングの効果を直接計る指標として、社外コンサルの指摘内容を指摘レベルで分類し、その件数の推移を計測した。指摘レベルは以下のように設定した。

A (応用) : さらに良くするための指摘 (would be better)

B (基本) : 修正すべき指摘 (should)

C (初級) : フォーマットやドキュメンテーションに関する指摘

計測結果を図5に示す。横軸はコンサルの実施月、縦軸は担当者2名が受けた合計の指摘件数を成果物数で割った値である。立ち上り時の指摘件数が少ない理由は、根本的な指導に時間を費やしていたためである。8月~11月は指摘件数が増加しているが、12月以降は指摘件数が減少している。これは低レベルの指摘(C)が減少しているためであり、同時に変更要求を抽出するプロセスや調査資料に関する指摘など、表2で設定した変更要求仕様書の指摘観点に収まらない高レベルな指摘の比率が増加しているためである。上記より、対象者の知識レベルが向上していると言える。

また、チェックシートの観点別に指摘内容を分類した。これより、網羅性に関する指摘、特に階層ごとの記述粒度に対する指摘が多かった。USDMでは、段階的に詳細化することによって抜け漏れの防止を図っているが、開発案件ごと

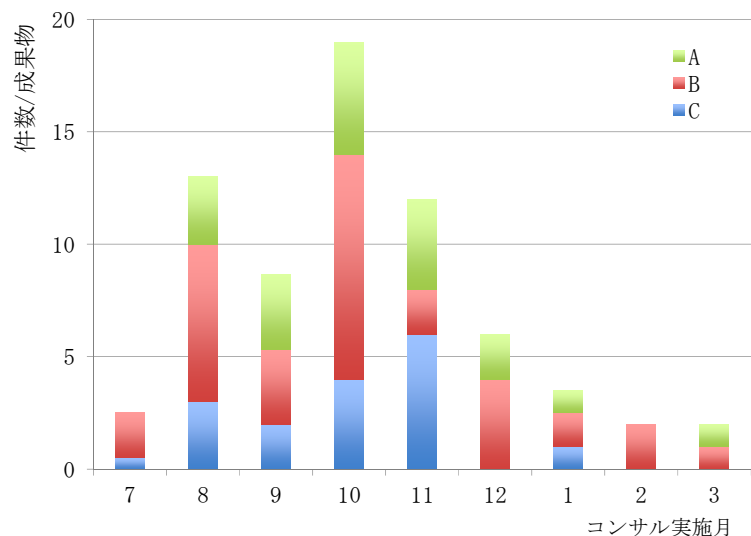


図5. コンサルでの指摘レベルの推移

に適切な記述粒度で記述することは一朝一夕では難しい。実開発の成果物を作成し、コンサルを繰り返し受けることによって、案件ごとに最適な粒度の USDM を記述するコツを習得できたと考えている。

5.3 レビューにおける指摘内容の推移 (Level3:Behavior)

次に、対象者が現場のレビューで適切に指導できているか、またそれにより現場の成果物の質が向上しているかを評価するために、レビューの指摘を以下の3つの領域に分類し、その件数の推移を計測した。

I 領域：効率性、保守性、移植性に関する指摘

II 領域：機能性、信頼性に関する指摘

III 領域：ドキュメンテーション、成果物の構成、ドキュメント不足に関する指摘

期待する効果として、成果物の質が向上することにより II 領域と III 領域の指摘が減少することを想定した。I 領域については、成果物の質が向上することで減少する一方、レビューの質が向上することで増加することが想定されるため、参考として計測するまでに留めた。結果を図6に示す。これより、II 領域の指摘が1割、III 領域の指摘が5割程度減少し、成果物の質向上が確認できた。

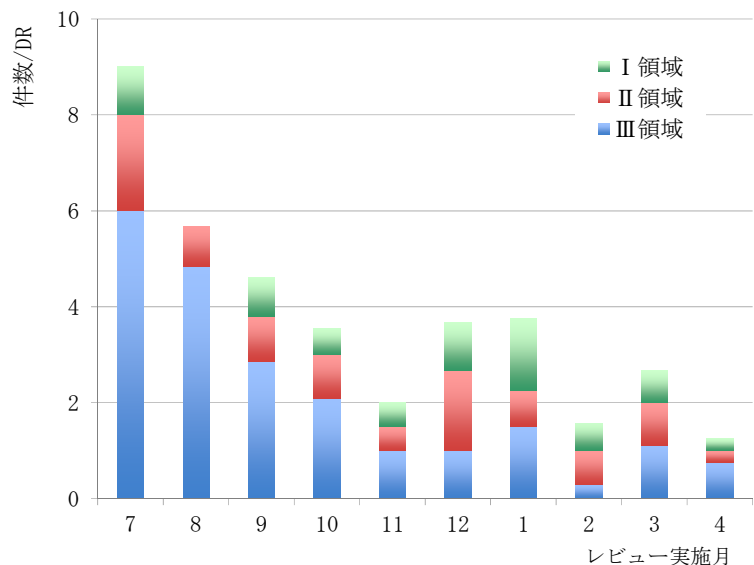


図6. レビューにおける指摘内容の推移

5.4 工程別の不具合摘出件数の推移 (Level4:Results)

最後に、本手法を行うことによる品質状況の変化を計測した。具体的には、検査工程で摘出された不具合件数を前半5ヶ月(7月~11月)と後半5ヶ月(12月~4月)で比較した。不具合件数は開発規模(ソース行数)で正規化している。計測結果を図7に示す。

結果、前半に比べ後半の不具合摘出件数は約1/2に減少した。本データは開発の難易度・リソースの質・納期の余裕度などの外部要因が影響している可能性があるが、図6でII領域(機能性・信頼性)の指摘が1割しか減少していないにも関わらず、不具合件数が1/2に減少していることから、変更要求仕様書の質が向上した効果が表れていると考えることができる。

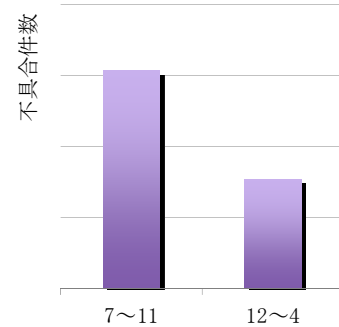


図7. 品質状況の推移

5.5 活動全体の考察

レビューでは、III領域の指摘が減少することにより、レビューの雰囲気がよくなったという副次的な効果があった。機能不具合につながらないドキュメンテーションに関する指摘は、指摘する側にとっても指摘される側にとっても心象がよくない。これらの指摘が減少することにより、レビューで活発な指摘が生まれやすい雰囲気を醸成することができたと感じている。

また、モデレータの導入により、レビュー自体の効果を高めることもできた。従来は、レビューア中心にレビューを進行していたものの、明確な進行役が決まっておらず、本質的ではない議

論に時間をとられる、本来審議すべき箇所が十分に審議されない、といった問題があった。また、成果物のリーディング方法も担当者に依存していた。例えば、変更要求仕様書では、変更使用を階層的に記述するため、階層ごとに仕様の抜け漏れをチェックするリーディング方法が効果的である。今回は、成果物の作成者以外がモデレータとしてレビューを進行することにより、第3者の観点で不具合が入り込みやすい部分や、成果物の記述がわかりにくい部分を中心にレビューすることができた。

今回、教育・業務の両面で第3者視点の観点を取り入れた対策を実施した。しかし、常に第3者視点を取り入れる活動を実施することはリソースの面から難しいと思われる。教育においては、教える側に知見・スキルが十分備わっていれば、第3者視点でなくても同様の効果を得ることは可能であると思われる。ただし、業務では、上下関係による非論理的な指摘やコスト・納期のしがらみによる妥協をさせないためにも、第3者視点による指導が重要である。例えば、SEPGやSQAが本責務を担うことができると考えている。

本活動は、選出した若手リーダーの上司・組織を巻き込み、オフィシャルな活動として実施した。教育は短期的に効果が現れにくく、定量的な評価が難しいため、活動の優先順が低くなりやすい。そのため、通常業務が忙しくなると、それを理由に本活動が滞ることが想定された。そこで、活動の内容や結果を上司に報告する報告会を開催することで、選出した若手リーダーが、通常業務を理由に本活動への参加が滞ることがないようにした。このように、第3者によるフォローアップの体制・仕組みの構築を行うことが、教育を運用するうえで重要であると考えている。

6. まとめ

XDDPのねらいや効果を理解し、実開発で実践できる技術者を育成するために、教える立場と教えられる立場を反復して繰り返すトレーニング手法を提案し、施行を行った。その結果、対象者の実践スキルの向上と組織的な成果物の質の向上を確認できた。しかし、成果物の質の向上は1割程度に留まっており、対象者以外のメンバの実践力は十分とは言えない。今後は、対象者以外の実践力を向上させるため、トレーニング手法の継続実施と更なる改善を行っていく予定である。

本手法の課題として、育成対象者をいかに選別するかがある。定期的なコンサルへの参加・担当外モジュールのレビューへの参加を行うため、対象者には相応の負荷がかかる。そのため、本手法は全メンバのレベル底上げに適用するのではなく、技術リーダーとして育成すべきメンバを選別して、集中してレベルを引き上げるのがよいと考えている。

なお、本手法はXDDPの実践力向上だけでなく、他の開発手法やプロセスの実践力を上げる目的でも適用できると考えている。今後は、急速に変化する市場状況に対応するために、本手法を用いて新たな開発手法・プロセスを習得したコア人材を育成していきたい。

参考文献

- [1] 清水吉男, 「派生開発」を成功させるプロセス改善の技術と極意, 技術評論社, 2005
- [2] 清水吉男, 要求を仕様化する技術・表現する技術, 技術評論社, 2010
- [3] 吉田新一郎, 効果10倍の<教える>技術, PHP研究所, 2006
- [4] Kirkpatrick, D.L., Evaluating Training Programs: The Four Levels, Berrett-Koehler Publications, Inc., 2005