

第3分科会（空）

HDR 法：仮説駆動型レビュー手法の提案 —HDR 法の実践による生産性と品質の同時向上— Proposal of HDR method. (HDR: Hypothesis Driven Review)

The best practice to adopt the HDR method to improve both productivity and quality.

主査	細川 宣啓	日本アイ・ビー・エム(株)			
副主査	永田 敦	ソニー(株)			
	藤原 雅明	東芝ソリューション(株)			
アドバイザー	森崎 修司	(国)静岡大学			
研究員	上田 裕之	(株)D T S	高橋 功	ソーバル(株)	
	高橋 実雄	(株)サンモアテック	中谷 一樹	T I S (株)	

研究概要

本研究チームでは、レビューアの思考メカニズムに着目し、時間制約の中でも集中的に重大欠陥を検出できるレビューア（以降、トップレビューア）の思考メカニズム解明を研究対象とした。

レビュー時の思考メカニズムを解明した結果、トップレビューアはレビュー対象および品質要求・品質特性から欠陥の兆候を掴み、兆候を基に仮説を立てて狙いを定めてから検証を行っていることが判明した。本研究では、トップレビューアが普段から行う暗黙知としての思考メカニズムを、手法として形式知化し初心者／未経験者でも実践できるように体系化した。

本手法を用いた実験により、効率的で効果的なレビューの実施、レビューアのスキル向上を図ることが可能になるという確信を得た。

Abstract

Our research team focused on reviewers' thinking mechanism and above all, researched the top reviewers' which helped them detect critical defects intensively even with a time restriction.

By clarifying the thinking mechanism, we found that they grasp a sign of defects from review objectives or quality requirement and quality characteristics, then focus and verify it after building up a hypothesis based on the sign of defects.

In this research, we formalized the tacit about quality, that is, thinking mechanism that they usually apply as a method, and then defined the process to enable beginners or inexperienced people to practice it. The experiment using this method gained us confidence that it enabled to carry out an effective and efficient review and to improve reviewers' skill.

1. はじめに

1.1 研究の背景とレビューに関する問題点

近年のソフトウェア開発において、レビューはソフトウェアの欠陥を早い段階で検出できる手段として、品質向上に寄与するだけでなく、コスト削減、納期短縮に有用である。しかし、レビューは属人性が高く、短時間で重大欠陥を検出できるいわゆる「トップレビューア」にレビュー依頼が集中する傾向がある。一般的にトップレビューアはプロジェクト内で多くの責務を担う場合が多く、十分なレビュー時間を取れない。そこで負荷集中の対策として他のチームメンバー等にレビューを任せることになるが、時間をかけても軽微欠陥しか検出できず、結局重大欠陥が残存し大きな手戻りや本番障害を引き起こすことがある。原因分析を行っても最終的に原因は人となるため、トップレビューアによるレビューの重要性が誇張され、更なる負荷集中を招くことがある。

一方、近年のソフトウェアは更に規模が大きくなり複雑化している。ソフトウェアの複雑度が増すとレビューで確認すべきポイントは、その組み合わせを考えると指数関数的に増加するため、今までのレビューのやり方で網羅度を上げていくだけではとても追いつかない。よって、今後は時間制約がある中でも重大欠陥だけは取り残さないようにする方法が必要と考えた。

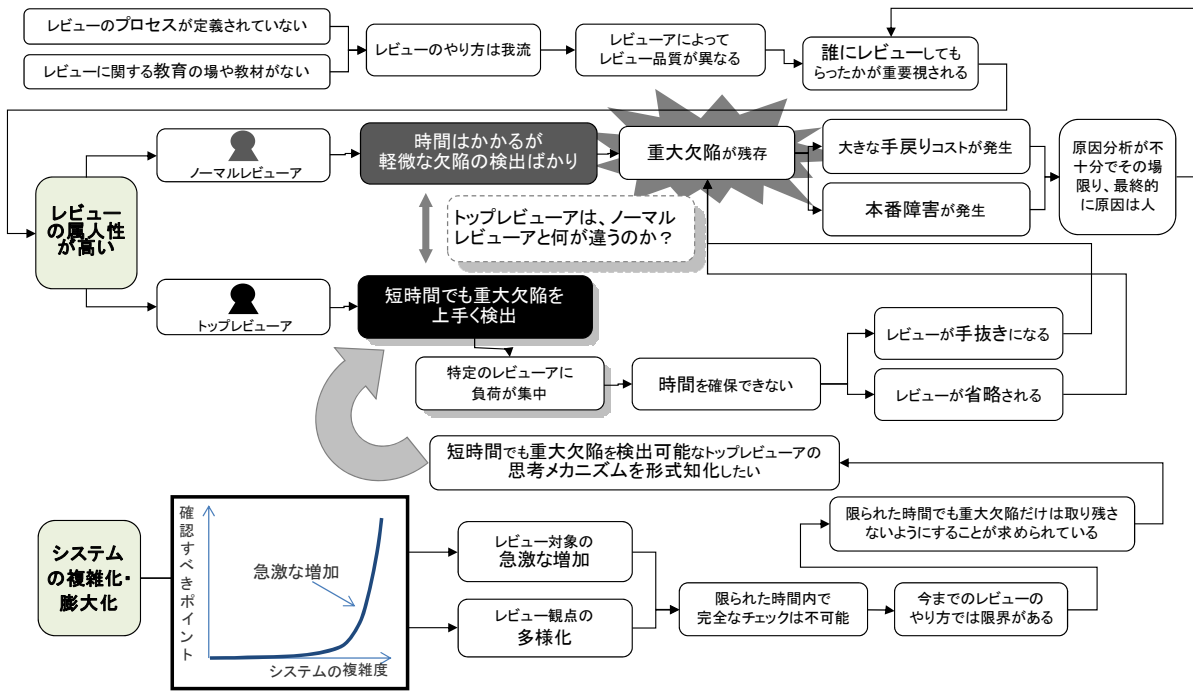


図 1-1 レビューで抱える問題の因果図

1.2 研究の狙い

我々は、時間制約がある中でも集中的に重大欠陥を検出できるトップレビューアの欠陥検出メカニズムを研究の対象とし、トップレビューアがレビュー時に何を見て何を考えているのか、その思考メカニズムを解明し形式知化して新たなレビュー方法を確立した。これにより他者へのレビューの技術移転を容易にし、レビュースキルを向上できるようになること、また誰でも時間制約がある中で集中的に重大欠陥を検出できるようになることが期待できると考えた。

2. 提案する欠陥検出の方法

本章では、我々の研究グループが提唱する「HDR法（Hypothesis Driven Review Method：仮説駆動型レビュー手法）」について記述する。

本手法を構成する要素である「思考プロセス」と、本手法を使いこなすための「必要な知識」について説明する。

2.1 HDR法の思考プロセス

研究の過程では、様々なプロジェクトにおいてレビューによる大きな成果を上げているレビューアの行動や考え方を観察した。その結果、各組織のトップレビューアに共通した思考プロセスがあることがわかった。それを図2-1に示す。

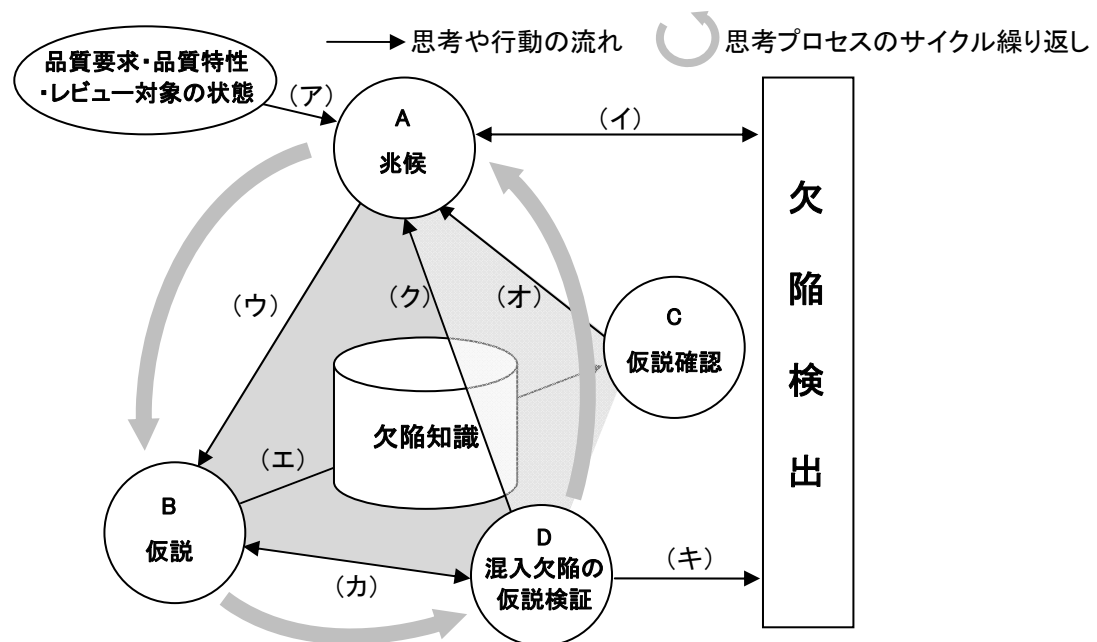


図2-1 HDR法概念図

以下に思考プロセスの詳細を示す。（具体例については「別紙6兆候-仮説例」を参照のこと）

2.1.1 兆候を掴む

レビュー時の第一の行動として、レビュー対象の中身を詳細に見るのではなく、兆候を掴むことから開始する。兆候とは、欠陥の存在を示唆する定量的・定性的なデータやキーワードなど、レビュー対象の内外に観察可能な「特徴」のことである。

例えば、ドキュメント内に「、や。（句読点）」と「，や。（カンマ、ドット）」が混在しているという特徴は、作成者以外の人間が後から追加した箇所があることを示唆する兆候となる。また、レビュー対象の作成時間（ファイルプロパティの作成日時から最終更新日時までの間隔）が極端に短いという特徴は、レビュー対象を急いで書き上げ、自己レビュー不足の状態であることを示唆する兆候となる。後者はレビュー対象の外側から掴むことができる兆候である。

更に、納期要求や動作環境制約などの品質要求・品質特性（後述）をインプットにして兆候を掴むこともできる(A)。兆候と紐付けられる欠陥は一つではなく、N対Nの関係となる。

なお、明らかな欠陥として表出している兆候は直接欠陥検出に結びつく(I)。この場合、その欠陥自体が他の欠陥の兆候となりうる可能性がある。

2.1.2 仮説を立てる

2.1.1 で掴んだ兆候から、混入欠陥の種類や位置、及び混入状態の仮説を一つ以上立てる(ウ)。仮説への根拠となる兆候も一つではなく複数の兆候を基にすることが多い。

通常、故意に欠陥を混入させる人はおらず、無意識のうちに欠陥は混入する。そこには人的ミスが発生しやすい状況、ミスを検知するプロセスの欠落、開発者同士の認識の齟齬、確認プロセスの形骸化など、何か原因がある。すなわち原因と結果の因果関係がある。

仮説を立てる際は、兆候を結果と捉えて、その結果が何故起こったのか、原因を推測する。または、兆候を原因と捉えて、その原因により何が起こるのか結果を推測する。

2.1.1 の例の場合、「句読点とカンマ、ドットの混在」という兆候を結果と捉えると「作成者以外の人間が後から追記」が原因の一つとして推測できる。更にその推測を原因として捉えると、その前後で不整合が発生しているという仮説を立てることができる。

2.1.3 仮説確認

品質要求・品質特性や2.1.2 で立てた仮説について、その場ですぐに確認可能なものは作成者やレビューイに確認を行う(エ)。この確認により、開発担当者の懸念事項や現場の不安／リスクを抽出でき、重点レビュー項目、重点レビュー観点が確定することになる。また、立案した仮説が事実として扱えるようになり(オ)レビューにおける仮説の確度が増すことになる。

2.1.4 検証する

2.1.2 で立てた仮説を検証するための観点を設定し(カ)、その観点到絞って実際にレビュー対象を検査する。その結果、欠陥を発見したり(キ)、新たな兆候を発見する(ク)。

レビュー中は、以上の2.1.1～2.1.4の思考プロセスのサイクルを回し続ける。しかし、必ずこの順番で行われるものでもない。例えば図2-1中「A」の兆候から即欠陥の発見に至ることや(イ)、「D」の検証を終えた後「B」の仮説立案へ戻り、別の仮説を立てて「D」の検証を再度行うこともある(カ)。なお、本手法はレビュー対象や見るべき観点を絞るプロセスであるとも言えるため、レビュー対象物の数や量が多くどこから手をつけて良いかわからないような場合、より高い効果を発揮する。ただし、観点を絞るため、レビュー結果における「指摘件数」は減る可能性がある。

2.2 HDR法で必要な知識

2.2.1 品質要求・品質特性

レビュー対象物を作成する上での構築環境や前提条件・制約条件といった外部要因や、レビュー対象物の直接的な品質を示す要素のことである(詳細は「別紙2 品質要求・品質特性例」を参照)。例えば、プロセス因子の中の一つの「納期」の情報をレビュー時に知ることができた場合、レビュー対象の仕様書のボリュームと併せて「兆候」の一つとなりうる。つまり、その納期が短納期であるにもかかわらず、仕様書のボリュームが非常に大きい場合は、複数名による同時作成という仮説が立てられる。その仮説が正しいと仮定した場合、担当者間の調整不足が起因となるインタフェース設計エラーなど仕様不整合の欠陥が混入しているという仮説を立てることができる。このように品質要求や品質特性は、兆候の一つとしてまたは仮説を立てる上での直接的な根拠として参考になる場合があり、仮説立案のための重要な情報と言える。

2.2.2 欠陥知識

本手法における思考プロセスにおいて、欠陥知識は重要な要素の一つである。一般的に欠陥知識と言うと指摘一覧票に記載された欠陥そのものを指すが、多くの欠陥を知っていればレビューでの欠陥検出率が上がるわけではない。欠陥そのものだけでは有効な知識とは言えない。トップレビューアは、何故その欠陥が混入したのか、その欠陥が混入した際に何か特徴的なことはなかったか（兆候）、どうやって検出できたのか（仮説）、それを十分に分析し、その兆候、仮説とセットで欠陥を覚えている。その欠陥知識に基づいた仮説パターンを多数持つておくことで大きな効果を発揮する。トップレビューアは、兆候把握から仮説立案・検証のサイクルを短い時間で何回も繰り返しており、その中では兆候－仮説をセットにして蓄積した欠陥知識とのマッチング処理を行っている。

なお、欠陥知識の記憶方法には様々な方法が考えられる。欠陥知識の記憶方法例を「別紙7 欠陥知識の記憶方法例」および以下に、いくつか紹介する。

表 2-1 欠陥知識の記憶方法例

タグ型	階層構造型	ベン図型
この兆候には、この欠陥というように、兆候に対してタグを付ける。	ツリー図などで、兆候→仮説を図化する。	複数の兆候から一つの仮説を立てられるようなものをベン図にする。
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> 【兆候】 COBOL 使い </div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> 【仮説】 異常系の記述漏れ </div>	<div style="text-align: center; margin-bottom: 5px;"> 【兆候】 フッタのフォントが違うページがある </div> <div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid black; padding: 5px; width: 45%;"> 【仮説】 類似システムの仕様書を流用した </div> <div style="border: 1px solid black; padding: 5px; width: 45%;"> 【仮説】 違う人が後から追加した </div> </div> <div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid black; padding: 5px; width: 45%;"> 【仮説】 類似システムで不要だったため今回必要な機能が書かれていない </div> <div style="border: 1px solid black; padding: 5px; width: 45%;"> 【仮説】 他との記述レベルが違う他との整合が取れていない </div> </div>	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; border-radius: 50%; padding: 10px; text-align: center; width: 30%;"> 【兆候】 キーワード「場合」が多い </div> <div style="border: 1px solid black; border-radius: 50%; padding: 10px; text-align: center; width: 30%;"> 【仮説】 条件漏れ </div> <div style="border: 1px solid black; border-radius: 50%; padding: 10px; text-align: center; width: 30%;"> 【兆候】 日本語の平文が多く図・表がない </div> </div>

3. 評価と結果

3.1 実験方法

実験では従来の自由なレビュー方法、仮説駆動型でレビューした場合の結果を比較し、HDR 法の有用性を評価した。レビュー対象として二つの要求仕様書、仕様書 X「アンケート記入システム」、仕様書 Y「ヘリコプター予約管理システム」をそれぞれ 5 ページ、約 2000 字の分量で自作した。研究員が所属する会社の中から所属部門、経験年数など、それぞれ立場の異なる被験者を選抜し、レビューを 2 回実施した。時間的制約のある状況下を想定し、レビュー時間は指摘記述も含めて 25 分間とした。

1 回目は通常のレビュー方法で実施した。その後、HDR 法に関する説明を行った後、2 回目を実施した。本手法には欠陥知識を要するため、説明の際は HDR 法による欠陥検出の実例を説明することで被験者に欠陥知識をインプットすることとした。なお、被験者のスキル、およびレビュー対象の難易度による偏りを排除するため、被験者を 2 グループに分けて実施した。（表 3-1 参照）

表 3-1 実験方法

	A グループ(6 名)	B グループ(6 名)	レビュー手法	レビュー時間
1 回目	仕様書 X	仕様書 Y	自由	25 分(指摘記述時間含む)
～ HDR 法に関する説明 ～				
2 回目	仕様書 Y	仕様書 X	HDR 法	25 分(指摘記述時間含む)

レビュー終了後、被験者には定量評価 5 項目、定性評価 3 項目からなるアンケートに答えてもらった。アンケート内容は表 3-2 の通り。

表 3-2 アンケート項目

定量	Q1.時間効率が上がりそうと感じたか.	定性	Q6.効果的と感じた部分について記載.
評価	Q2.重大欠陥を検出できそうと感じたか.	評価	Q7.実施する際に問題と感じた部分について記載.
項目	Q3.スキル向上に役立ちそうと感じたか.	項目	Q8.その他, この手法に関する意見・感想について記載.
	Q4.通常のやり方と違うと感じたか.		
	Q5.現実的ですぐに実施可能な方法と感じたか.		

定量評価においては、項目に対して「5.非常に有益だと思う」「4.まあまあ有益だと思う」「3.どちらでもない」「2.あまりそう思わない」「1.全くそう思わない」の5段階評価、定性評価は設問内容に対して意見を自由記入とした。

実験終了後、アンケートおよび1回目と2回目の欠陥指摘リストを被験者より回収した。

3.2 評価方法

重大欠陥を「システムテスト工程で見つかった場合、手戻り工数が8h以上かかると予測される欠陥」、「ビジネスニーズを満たせない欠陥」、「テストで検出できない欠陥」の3種類の欠陥に分類し1回目と2回目における欠陥指摘数の推移を比較した。さらに、アンケート結果から効果の程と改善点を得る。以上の内容を総合的に評価し、HDR法の効果を測定した。

3.3 実験結果

被験者12名から回収した欠陥リスト(有効件数12件)から次の結果が得られた。

3.3.1 欠陥指摘数

欠陥指摘数平均は、1回目(自由手法)7.08件に対して、2回目(HDR法)5.92件となり、HDR法の方が欠陥指摘数は少なくなるという結果となった。

表 3-3 欠陥指摘数平均

N=12			
	1回目 指摘数	2回目 指摘数	比率
欠陥指摘数	7.08	5.92	16% down

3.3.2 重大欠陥の指摘数

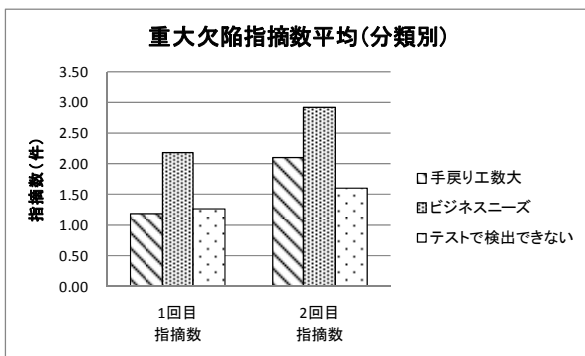


図 3-1 重大欠陥指摘数平均グラフ (分類別)

重大欠陥の平均数は1回目に対して2回目の方が増加する結果となった。

「手戻り工数8h以上」、「ビジネスニーズを満たせない」、「テストで検出できない」の重大欠陥指摘数平均については、それぞれ79%、35%、27%増加しており、HDR法が重大欠陥検出に有効であることを示唆する結果となった。

表 3-4 重大欠陥指摘数平均

N=12			
	1回目 指摘数	2回目 指摘数	比率
手戻り工数大	1.17	2.08	79% UP
ビジネスニーズ	2.17	2.92	35% UP
テストで検出できない	1.25	1.58	27% UP

3.4 アンケート結果

被験者から回収したアンケート結果は以下の通りとなった。

3.4.1 定量評価結果

表 3-5 定量項目アンケート結果

設問	5	4	3	2	1	賛同 (4以上)	賛同 割合
Q1.時間効率が上がりそうと感じたか？	3	5	1	3	0	8	67%
Q2.重大欠陥を検出できそうと感じたか？	2	6	2	2	0	8	67%
Q3.スキル向上に役立ちそうと感じたか？	5	4	2	1	0	9	75%
Q4.通常のやり方と違うと感じたか？	3	6	2	1	0	9	75%
Q5.現実的ですぐに実施可能な方法と感じたか？	2	3	2	4	1	5	42%

「5.非常に有益だと思う」「4.まあまあ有益だと思う」「3.どちらでもない」「2.あまりそう思わない」「1.全くそう思わない」

設問1～3において、4以上（賛同）が67%以上となり、時間効率向上、重大欠陥検出、スキル向上の効果があると思う被験者が3分の2以上という結果となった。設問4については、4以上（通常のやり方と違う）が75%という結果となった。設問5については、4以上（現実的ですぐに実施可能）が42%という過半数を割る結果となった。

3.4.2 定性評価結果

傾向としては以下のような評価が多かった。

- 兆候に着目することで全体を俯瞰することができるようになる
- 重大な欠陥を効率よく見つけることができる手法であると感じられた

また、課題として以下のような評価を得られた。

- 兆候から仮説を導く部分が難しく、知識や慣れが必要

4. 考察

前章の実験結果から、レビュー対象が大規模、または時間制約がある中でレビューしなければならない条件下、および欠陥知識が十分に身につけられた条件下においては有効な手法であることの確信を得た。実験結果とアンケート結果に関する考察を以下に記述する。

4.1 実験結果に関する考察

既に3.3章で述べた通り、HDR法を用いた方が、総欠陥指摘数は減少したが、時間制約の中で重大欠陥の指摘数は向上した。統計的な有意差こそ見られなかったが、（統計分析結果は別紙8を参照）一部有意性は見られた。その要因として以下が考えられる。

- 本手法に不慣れなため、兆候を探すまたは兆候から仮説を立てる作業に手間取ったため、総欠陥指摘数は少なくなった。
- 兆候を探す過程で、より重大な欠陥を探そうという意識が働くため、軽微欠陥の指摘数が少なくなり重大欠陥の指摘数が向上した。

4.2 アンケート結果に関する考察

現場視点での有効性評価・課題抽出を目的として実施したアンケートにおいて、既に3.4章で述べた通り、定量評価項目のうち、時間効率向上、重大欠陥検出、スキル向上の3項目全てが高

い評価であった。ただし実施可能性の評価は高いとは必ずしも言えない結果であった。この点は定性評価項目では、「兆候から仮説を導く部分が難しく、知識や慣れが必要」という意見とも整合するが、全体的に「重大欠陥を効率よく検出できる手法」、「レビュー対象が膨大な場合に有効な手法」という意見を多く得られた。

定量評価・定性評価を総合評価すると、手法の導入初期は欠陥知識不足や仮説的思考の不慣れが原因で欠陥検出効率が上がらないが、経験を積み知識を蓄積すれば効率的・効果的なレビュー実施が可能な手法であると評価されたと言える。換言すると、本手法は、従来経験者が少なく体系的に実施されていなかった思考プロセスを用いていること、および継続的实施によるレビューアのスキル向上により生産性・品質の同時向上が可能な手法であるということである。

本手法の導入初期に効率が上がらないという課題については、適切な「欠陥知識移転の介助」と兆候-仮説-検査観点のパターン共有等を通じた現場適用時の支援充実が必要である。あくまで小さな単位で、こまめな兆候と欠陥の記録を継続することで、長期的には「欠陥」や「兆候」等の資産化が実現できる。一度目のレビューから結果を出すことは難しいものの、HDR法という知識のフィードバックサイクルに「欠陥」という知識を乗せて蓄積／共有・移転を行うことで、レビューの検出速度・精度の両面で期待できると考える。

5. まとめ

本研究では、トップレビューアの欠陥検出メカニズムに着目して、兆候を基にした「HDR法 (Hypothesis Driven Review Method: 仮説駆動型レビュー手法)」を提案した。本手法は、プロダクト因子やプロジェクト因子などの品質要求・品質特性やレビュー対象物そのものから欠陥の兆候を掴み、その兆候を基に仮説を立てて、狙いを定めて欠陥を検出する、効率的かつ効果的でレビューアのスキル向上にもつながる手法である。実験によって、本手法を用いることで時間制約がある中でも、重大な欠陥を集中的に検出できるようになることの確信を得た。

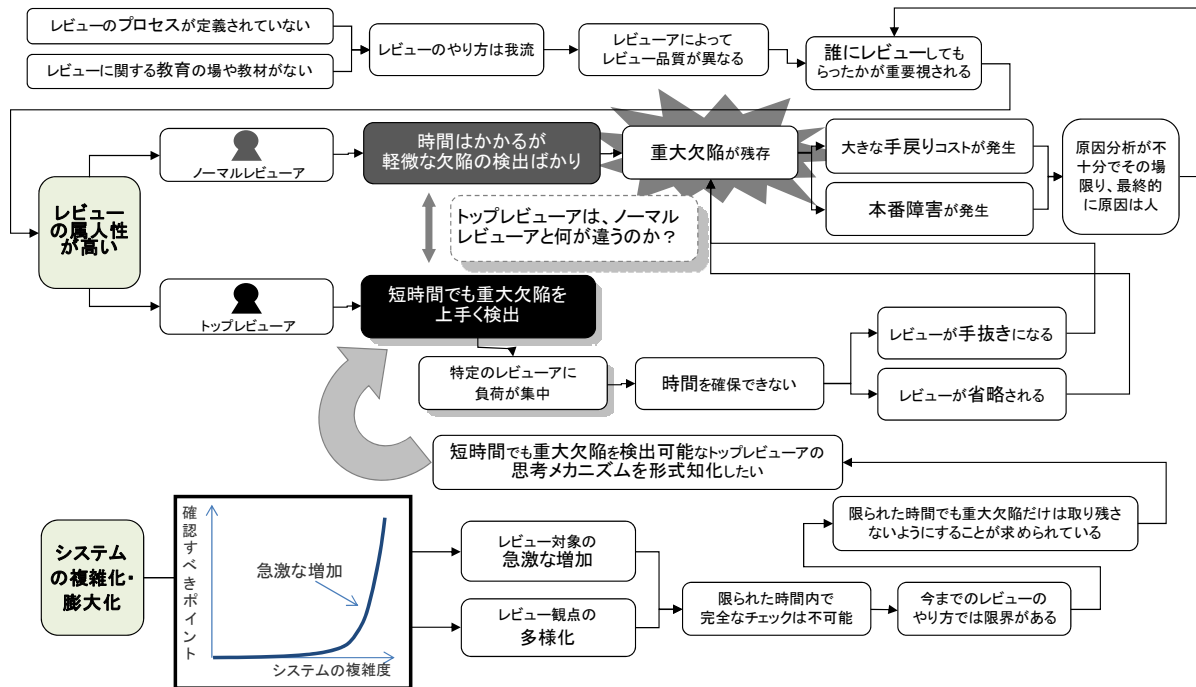
近い将来、システムの複雑化により完全網羅的なレビュー実施が困難になることを考えると、短時間でも重大な欠陥を検出可能な本手法を活用する場面は多くなると考える。そのためにもいち早く本手法を取り入れ、欠陥知識を蓄えレビュースキル向上を図り更なるトップレビューアを育てておくことが切望される。

参考文献

- [1] 「ピアレビュー」 日経BPソフトプレス社 Karl E. Wiegers (著)
- [2] 「ソフトウェア品質知識体系ガイド—SQuBOK Guide」 オーム社 SQuBOK 策定部会
- [3] 「医者現場でどう考えるか」 石風社 グループマン, ジェローム
- [4] 「考えることの科学—推論の認知心理学への招待」 中公新書 市川 伸一 (著)
- [5] 「認知心理学 知のアーキテクチャを探る」 有斐閣アルマ 道又 邇, 北崎 充晃, 大久保 街亜, 今井 久登, 山川 恵子, 黒澤 学

附録

別紙1 レビューで抱える問題の因果図



別紙2 品質要求・品質特性例

プロセス因子	
プロジェクトの安定性	
スコープの明確さ	
スケジュール	
納期	
設計ドキュメント作成時間	
制約・予定・進捗状況	
課題・リスク	
コスト	
制約・予定・進捗状況	
課題・リスク	
品質	
制約・予定・進捗状況	
課題・リスク	
ステークホルダ	
お客様の開発経験，業務理解度，権限	
経営層からのサポート	
課題・リスク	
人的資源	
作成者	
経験	
スキル	
立場	
マネージャー	
プログラマー	
制約・予定・進捗状況	
課題・リスク	
コミュニケーション	
人間関係	
開発体制	
自社 or 委託	
役割の明確さ	
課題・リスク	
調達	
契約内容	
制約・予定・進捗状況	
課題・リスク	

プロダクト因子	
要求の充足度，安定性	
機能要求	
非機能要求	
使用性	
信頼性	
効率性	
保守性	
移植性	
ビジネスリスク発生で重大障害・信用失墜	
ビジネス停止リスク	
得意先等重要利用者との関係悪化リスク	
法的リスク	
誤情報流出リスク	
経済的な大損失リスク	
社会インフラ停止リスク	
人命にかかわるリスク	
採用技術の安定性	
開発言語	
(技術の)成熟度	
技術的背景(歴史)	
ミドルウェア	
環境の充足度，安定性	
開発環境	
動作環境	
オンプレミス	
クラウド	
プロダクトの構造	
フルスクラッチ	
パッケージ	
レビュー因子	
レビュー計画の充足度	
目的	
観点	
ビジネスニーズを満たしているか	
後続のフェーズに渡せるか	
レビュー形式	

標準ルール・ガイドの遵守度・活用度	
開発工程	
	レビュー実施タイミング
	マイルストーン・チェックポイント
技術	
	ソースコード規約
	法律・規格
成果物	
	検収基準
開発ツール	
過去トラブルの活用	

対象	
開始基準／終了基準	
評価方法	
スケジュール	
事前準備	
	レビュー対象を受け取った際の状況
レビュー実施	
	集合会議の雰囲気
レビュー振り返り	

別紙3 実験結果 一定量データ

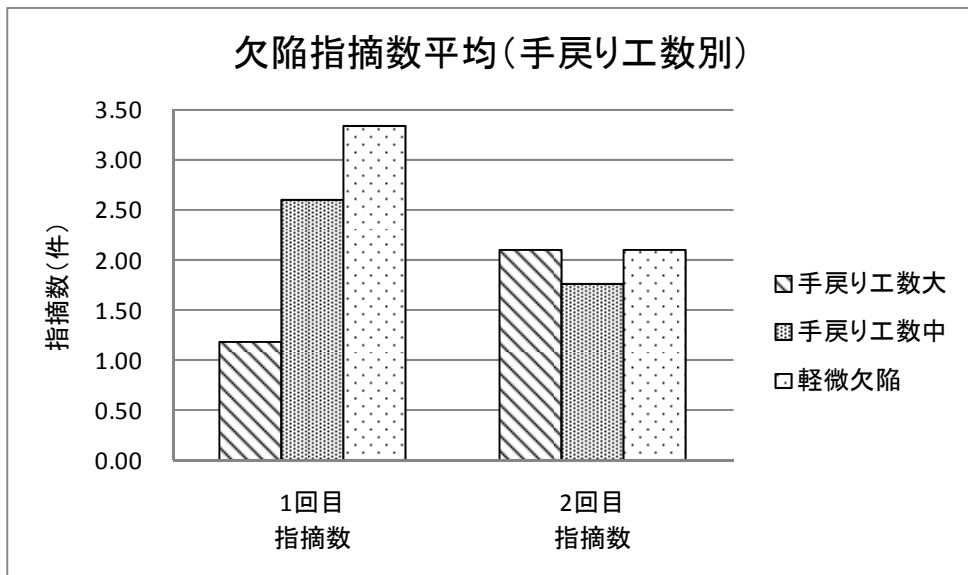
(1) 実験結果 (定量データ詳細)

被験者						1回目(自由手法)					2回目(仮説駆動型レビュー)					順		
氏名	所属部門	業務上の立場	レビュー頻度	部門経験年数	IT業界経験年数	1_総検出欠陥数	1_手戻り工数大	1_手戻り工数中	1_軽微欠陥	1_ビジネスニーズを満たせない	1_テストで検出できない	2_総検出欠陥数	2_手戻り工数大	2_手戻り工数中	2_軽微欠陥	2_ビジネスニーズを満たせない	2_テストで検出できない	Xから実施=1
S01	品質	TE	年1	5	5	11	1	4	6	3	1	8	4	0	4	4	4	1
S02	品質	TE	年1	8	8	12	1	7	4	3	2	10	7	3	0	8	5	1
S03	品質	TE	年1	8	8	6	0	4	2	2	1	5	3	0	2	3	0	1
S04	品質	TE	月1	8	8	9	1	5	3	3	1	5	5	0	0	5	3	1
S05	品質	TE	月1	2	8	5	2	0	3	2	0	5	1	2	2	2	0	2
S06	品質	TE	月1	5	5	2	1	1	0	2	1	4	1	3	0	3	0	2
S07	品質	TE	月1	6	18	8	0	3	5	0	1	4	1	2	1	1	1	1
S08	品質	TE	年1	7	10	7	2	3	2	4	2	6	0	2	4	3	2	2
S09	品質	TE	なし	8	8	5	1	1	3	1	1	6	1	3	2	5	2	2
S10	開発	TL	週1	11	11	9	4	2	3	4	4	9	1	4	4	0	0	2
S11	開発	TE	なし	10	10	5	1	1	3	2	1	5	0	1	4	0	0	2
S12	開発	TL	週1	10	10	6	0	0	6	0	0	4	1	1	2	1	2	1
平均				7.3	9.1	7.1	1.2	2.6	3.3	2.2	1.3	5.9	2.1	1.8	2.1	2.9	1.6	

(2) 実験結果 (欠陥指摘数平均 (手戻り工数別))

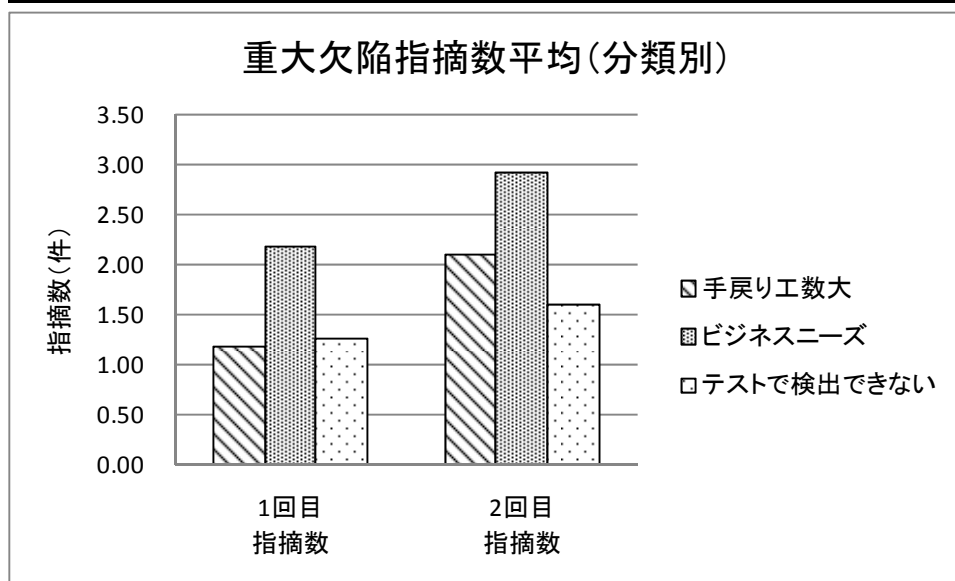
N=12

	1回目 指摘数	2回目 指摘数	比率	1回目 割合	2回目 割合	比率
手戻り工数大	1.17	2.08	79% UP	16%	35%	114% UP
手戻り工数中	2.58	1.75	32% down	36%	30%	19% down
軽微欠陥	3.33	2.08	38% down	47%	35%	25% down
総欠陥指摘数	7.08	5.92	16% down	-	-	-



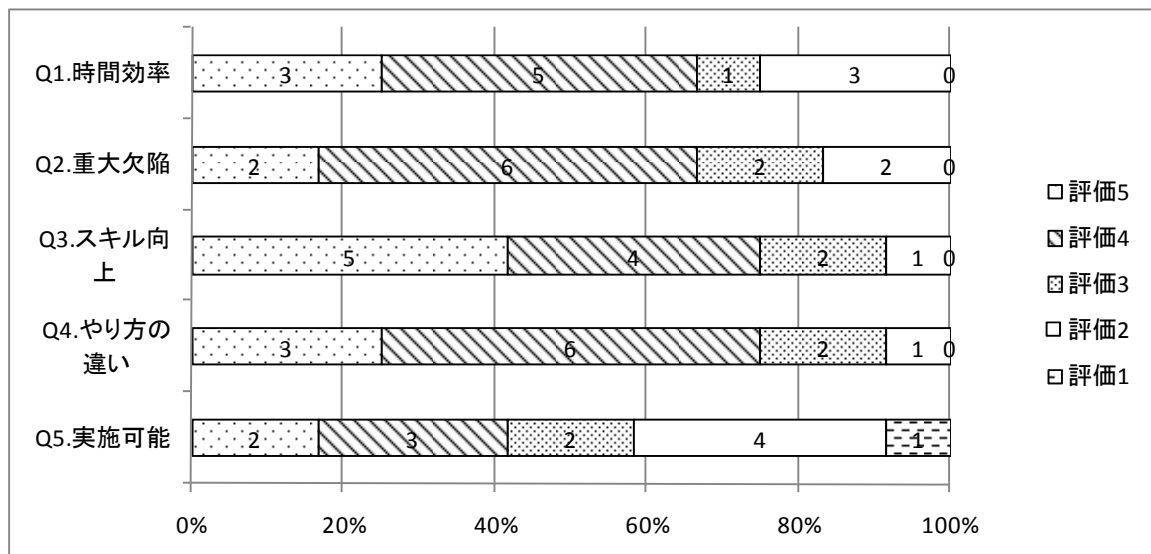
(3) 実験結果 (重大欠陥指摘数平均 (分類別))

	1回目 指摘数	2回目 指摘数	比率	1回目 割合	2回目 割合	比率
手戻り工数大	1.17	2.08	79% UP	16%	35%	114% UP
ビジネスニーズ	2.17	2.92	35% UP	31%	49%	61% UP
テストで検出できない	1.25	1.58	27% UP	18%	27%	52% UP



別紙4 アンケート結果 一定量データ

被験者						定量アンケート				
氏名	所属部門	業務上の立場	レギュラー頻度	部門経験年数	IT業界経験年数	Q1 時間効率	Q2 重大欠陥	Q3 スキル向上	Q4 やり方の違い	Q5 実施可能
S01	品質	TE	年1	5	5	5	4	5	4	5
S02	品質	TE	年1	8	8	2	4	3	3	3
S03	品質	TE	年1	8	8	2	2	2	2	1
S04	品質	TE	月1	8	8	5	5	5	4	5
S05	品質	TE	月1	2	8	4	4	4	5	4
S06	品質	TE	月1	5	5	4	5	5	5	4
S07	品質	TE	月1	6	18	4	3	4	4	2
S08	品質	TE	年1	7	10	4	3	4	5	3
S09	品質	TE	なし	8	8	5	4	5	4	4
S10	開発	TL	週1	11	11	2	4	5	3	2
S11	開発	TE	なし	10	10	4	4	3	4	2
S12	開発	TL	週1	10	10	3	2	4	4	2
平均				7.3	9.1	3.7	3.7	4.1	3.9	3.1



「評価5.非常に有益だと思う」「評価4.まあまあ有益だと思う」「評価3.どちらでもない」「評価2.あまりそう思わない」
 「評価1.全くそう思わない」

別紙5 アンケート結果 一定性データ

(1) HDR 法の利点

Q2. 効果的と感じた部分がありましたら記載願います。	
S01	「仮説を立てて、その仮説に基づいて情報を拾っていく」という部分が大変興味深いです。また、仕様書を記載した人物の連想や、同じ人が書いたか否かなど、全く考えたことがなかったです。目的意識がはっきりしていると効率的に作業できますね。
S02	文章を見て、欠陥の癖が分かるのは面白いと思いました。
S03	いくつかお話いただいた「兆候」については、実際のレビューで実践可能だと思いました。 ・はず、かも、など ・更新日付 ・章番号の重複
S04	誰が不幸になるのかという観点は、障害検出の優先度づけにもなると思いました。
S05	兆候は見つけることができることもあったが、兆候を掴むところから仮説をたて検証していくことで広げて行く方法を学ぶことができた。
S06	1度目と比べ兆候を見つけてることを意識した2度目は多く見つけたため、兆候を探すことが効率的なレビューにつながると思いました。
S07	例外処理や文字が多い所など欠陥が潜んでいるところを効率的にみられるところは効果的と感じました。
S08	文書を何人が担当してるかとかは意識したことがありませんでした。句読点や文書表現を想像して、そこから欠陥まで想像するというのは意識したことがないので、参考になりました。
S09	仕様書を読むと、どうしても細かいところに気が向いてしまい、大きな枠でとらえることを忘れてしまうため、「兆候をつかむ」という観点は非常に有効だと思いました。
S10	仮説に基づく深堀ができるため、重大な欠陥はより検出できるのではないかと感じる。
S11	重大な欠陥を検出しやすい探し方であるため、時間が少なくても効率よく欠陥を発見できる。
S12	兆候をつかむことにより、欠陥が混入しそうな部分の目安をつける部分が効果的だと思いました。

(2) HDR 法の課題

Q3. 実施する際に問題があると感じた部分がありましたら記載願います。	
S01	特にありませんでした。
S02	状況によっては実施できないと思われます。 例：お客様が仕様についての質問を望んでいない時、仕様書がない時など
S03	実際の現場で有効に適用できるようにするために、現場の問題点、課題にマッチした説明(教育)が必要なのかなと感じました。優秀なレビューを育てるために教育の観点で組織に推進させていく必要があると思いましたし、その結果、組織として効果を得られる内容だと感じました。
S04	特にありませんでした。
S05	この手法に慣れるまでは、効果を出せない
S06	新規の仕様書を流し読みすると、ほとんど内容が頭に入らず何度も読み直したり、部分的に集中して読んだりするため、仕様書を読む時間が増えたように思います。
S07	普段の手順書作成ではやはり全ての機能を理解したうえで方針に従ってチョイスすることが求められていると思うので、今回のやり方は現実的でないと感じました。時間に制限のあるレビュー的な立場の場合は有効だと思いました。
S08	ソフトウェア系の開発初期に携わる場合においてが最も有効かなと思いました。しかし残念ながらレビューを必要とされない仕事も多いのかと思います。
S09	この手法を実践する場合、ある程度の評価経験が必要だと感じました。 評価の経験が全くない人には、そもそも「兆候」が何かに気付くことができないと思います。
S10	兆候から仮説を導く部分には慣れが必要。 部分的なレビューになることが多そうで、満遍なくレビューをしたい場合などには向かない場合があるのでは？
S11	レビューや実践経験がある程度なければ欠陥の仮説を立てにくい。
S12	兆候をつかむのにはある程度のデータ量が必要だと思う。 データ量が少ない場合、兆候がつかみにくいと思う。

(3) その他, ご意見など

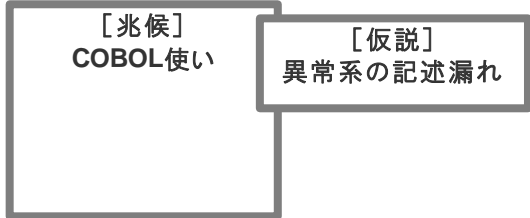
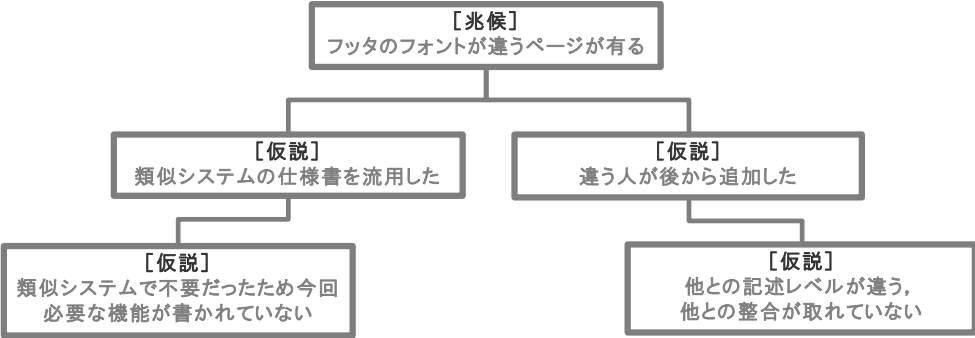
Q4. その他、この手法に関してご意見、感想がありましたら記載願います。	
S01	探索的テストのヒューリスティックテスト戦略モデル(仕様書を読む際に、これをベースにアクティブリーディングし、モデル化することを勧められていた)と併せてこの仮説駆動型レビューを行えば、より効果的なのかなと思いました(問題提起リスト&仮説リスト)。
S02	優秀なレビューアーのレビューを見てみたいです。
S03	社内のレビュー教育について「ぜひ推進してほしい(いきたい)！」という気持ちがあります。今日は、貴重なお話ありがとうございました。
S04	具体的な限られた時間でレビューをする事がないので、時間を区切るととても集中できるのだという事を実感しました。ありがとうございました。
S05	今後も勉強の機会をつくって欲しい。
S06	特にありません。
S07	現状の作業の中ではより粒度の細かい作業を求められているので、この手法では難しいと思います。時間が限られるような状況や大まかな仕様の定義のときなどに有効なように思いました。
S08	リリースタイミングで障害具合は予想しますが、個所まではなかなか思いつかないので今後担当さんの対応資料の熟成具合も参考にしたいと思えます。
S09	1回目のレビューでは誤字脱字に気を取られてしまい、全体を見渡すことができませんでした。 2回目では欠点の重さに気持ちが行ってしまい、どこまでどの順で指摘をしていいのかわからなくなりました。
S10	兆候から仮説を導くためのノウハウをためていくのが大変。 しかし一度ノウハウが蓄積されれば、非常に有用と感じる。
S11	1回目のレビューで、どのような観点でレビューを行うべきか書かれていないため、レビュー時に少し戸惑ってしまいました。
S12	膨大な量のデータのレビューには有効そうに感じた。 クロスチェックなどのレビューに適用はできないと感じた。

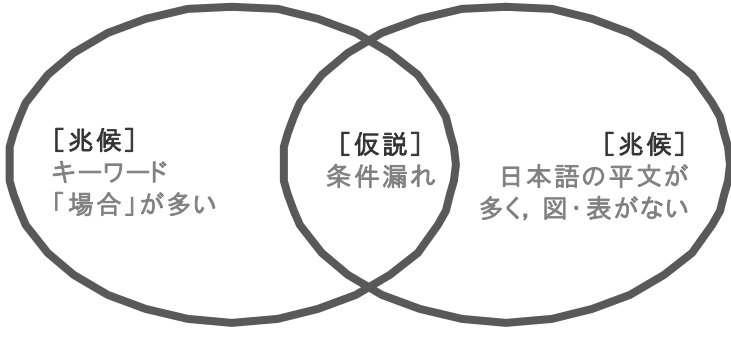
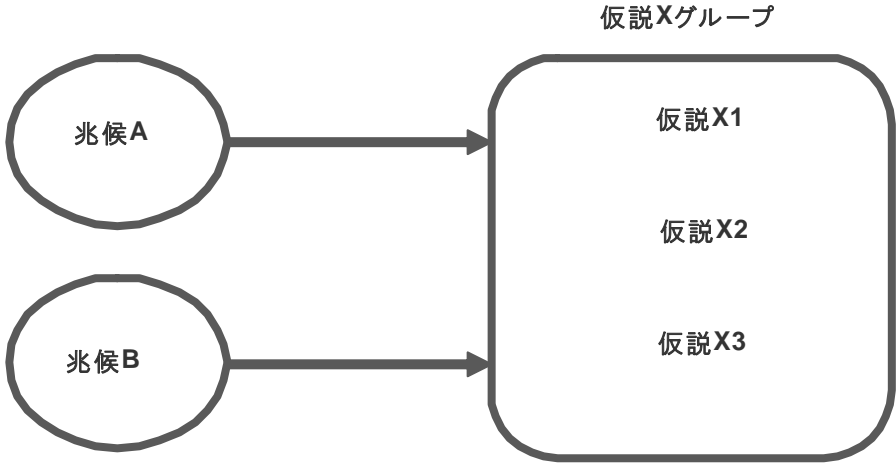
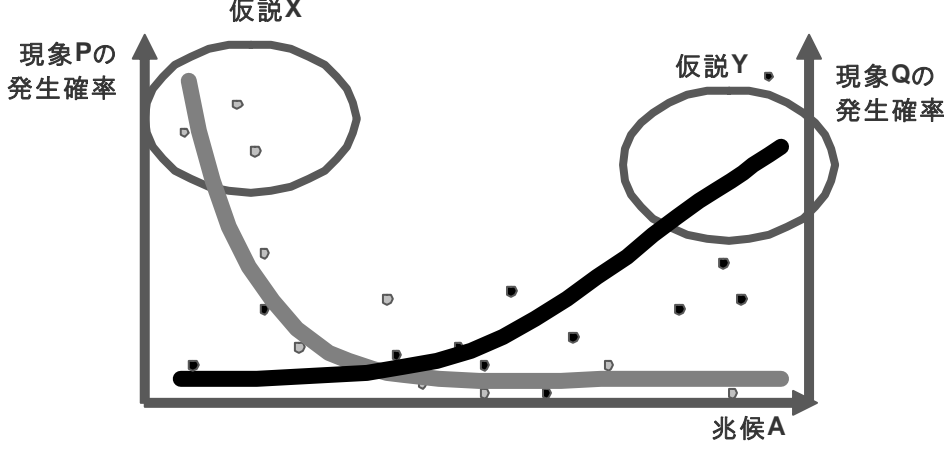
別紙6 兆候-仮説例

品質要求・品質特性・レビュー対象の状態	兆候	仮説	検証
<p>プロダクト因子</p>	<p>「現行通り」、「～のまま」、「～と同様」のキーワードが出現</p>	<p>全く同じにすれば良いと思ってしまう(現行の仕様通りでは現状の業務に合わない)</p>	<p>運用で対応していそうなところがないか確認(非効率な機能や使われていない機能までそのまま移植するのは無駄なのでその機能は移植しないという選択肢もある)</p>
		<p>既存システムの仕様は明確だと思ってしまう(既存システムの仕様が不明確なため実装できない・異なる仕様で実装してしまう)</p>	<p>「現行通り」というキーワードを検索する(何がどこまで同じなのか明確になっているか確認する)</p>
<p>ビジネス因子</p>	<p>固定でも問題なさそうなところだが柔軟性を求めている(社員IDは8桁固定ではなく拡張可能にするなど)</p>	<p>利用者数, 組織数, 拠点が増加</p>	<p>データ量や性能について要件が書かれているか(社員数等の増加を見込んでいるか)</p>
		<p>お客様の会社が近い将来合併するかもしれない</p>	<p>なぜ固定ではなく拡張可能にするのか理由が記載されているか確認する(会社の合併が理由であれば他の仕様変更の可能性についても確認する)</p>
<p>プロジェクト因子</p>	<p>WEBシステムで必須の内容が漏れている(認証機能, アップデート機能, 通信中の回線瞬断を考慮していないなど)</p>	<p>スキル・経験がない(WEB開発経験がない人が仕様書を記述)</p>	<p>WEB特有の画面遷移や操作に対応しているか(ブラウザの「戻る」ボタンが押された場合, ブラウザを閉じられた場合の後処理が書かれていない)</p>
<p>仕様書のファイルプロパティ情報</p>	<p>最終更新日付がレビュー前日の深夜</p>	<p>急いで作った</p>	<p>異常系の処理については省略されている, コピペして修正なし</p>

		誰にも相談せず1人で作り 上司のレビューも受けていない	真の要求を理解しておらず 開発者視点で書かれているためビジネスニーズを満たせない, 実業務に合わない仕様がある, 大事な要求が漏れている
	ファイルサイズが大きい	画像の貼り付け	既存の類似画面の画面コピーを流用, 画面作成後に仕様書を作成
仕様書の概観	日本語がぎっしりで図表は使われていない	日本語の曖昧さに起因する欠陥が含まれていないか	多義文:A または B でない場合, 全てテストできない
		条件が多い場合にある条件についての記載漏れがないか	表にすれば明確になるが文章だけだと漏れが分かりにくい
	句読点「,」「.」がなく長い文	自信がない, 人から聞いた内容をそのまま記載	根拠が正しく書かれているか, 誤解が無い
	似たような文書	コピペ	修正漏れ
	空白が多いページがある	何か記述する予定	仕様が未確定
		書いていたものを削除	文書構成の変更による不整合
フォントが違う	後から追記, 別の人が記入	他との整合性が取れていない, 修正漏れ	
仕様書内のキーワード	キーワード「はず」「かも」	お客様に確認していない, 本当か確信がない	-
	キーワード「場合」「時」の出現数が多い	キーワード「場合」が多いところは分岐が多く処理が複雑	異常処理の記述漏れ, 網羅的なテストができない
	キーワード「など」	洗い出しが不十分, 仕様未確定	-
検出した欠陥	章番号が重複している	後から章を追加した	追加した側の内容が別のページにも反映すべきだが漏れている
	章番号が飛んでいる	後から章を削除した	削除したことによる影響が十分考えられているか

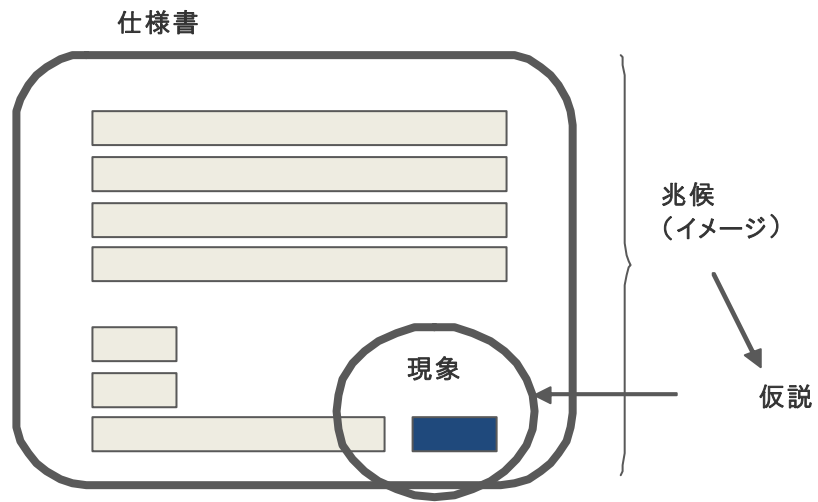
別紙7 欠陥知識の記憶方法例

種類	概要
タグ型	<p>この兆候が出たらこの欠陥というのがすぐに出てくるように、兆候にタグを付けて覚えておく。</p> 
ツリー構造型	<p>兆候→仮説のツリー図を作り覚えておく。</p> 
シナリオ型	<p>兆候を掴んで仮説を立てるまでの流れをシナリオに落として覚えておく。</p> <div style="border: 1px solid black; padding: 10px;"> <p><シナリオ> 複雑な仕様で文書表現が難しかったので、何度も推敲した文書には、変な助詞が残ることがある。 [兆候] 変な助詞「をは」、「がに」 [仮説] 仕様が難しく何度も推敲した [仮説] 仕様が完全に固まっているか、誤解が無いか</p> </div>
スクリプト型	<p>原因、兆候、仮説の情報を意味のある一つのスクリプトにまとめて覚えておく。</p> <div style="border: 1px solid black; padding: 10px;"> <p><異常系エラースクリプト> 現象：正常系しか記述されておらず、エラー時の処理が記載されていない 原因：書く時間が無かった、異常系を記載する癖がない 兆候：深夜に作成、COBOLの達人 良く出る場所：「場合」の近く 良く出す人物像：40代の元プログラマー</p> </div>

<p>ベン図型</p>	<p>複数の兆候を基に仮説を立てる場合は、ベン図にして覚えておく。</p> 
<p>グループ型</p>	<p>同じような仮説をグループにまとめて、そのグループに対する兆候をひもづけて覚えておく。</p> 
<p>グラフ型</p>	<p>兆候の大きさと、ある現象の発生確率を散布図などのグラフで表して、イメージで覚えておく。</p>  <p>例：仕様書の総ページ数が少なすぎる場合、仕様漏れ・仕様曖昧の可能性大 逆に、多すぎる場合、設計までされている可能性大</p>

映像記憶型

ある現象を見つけた時のイメージ（兆候）をそのまま覚えておく。



別紙8 実験結果の統計分析結果 – 定量データ

(1) 基本統計量及び正規分布適合検定結果

		1回目					2回目				
		総欠陥数	手戻り工数大	ビジネスニーズを満たせない	テストで検出できない	仕様書に書いていない	総欠陥数	手戻り工数大	ビジネスニーズを満たせない	テストで検出できない	仕様書に書いていない
分位点	最大値	12	4	4	4	6	10	7	8	5	9
	四分位点	9	1.75	3	1.75	5.75	7.5	3.75	4.75	2.75	4
	中央値	6.5	1	2	1	3	5	1	3	1.5	3
	四分位点	5	0.25	1.25	1	2	4.25	1	1	0	2
	最小値	2	0	0	0	0	4	0	0	0	0
基本統計量	平均	7.0833333	1.1666667	2.1666667	1.25	3.4166667	5.9166667	2.0833333	2.9166667	1.5833333	3.0833333
	標準偏差	2.8431204	1.1146409	1.3371158	1.0552897	2.1087839	2.0207259	2.1933094	2.3532698	1.7298625	2.3915888
	平均の標準誤差	0.8207382	0.3217691	0.3859921	0.3046359	0.6087535	0.5833333	0.6331539	0.6793305	0.4993683	0.6903922
	平均の上側95%	8.8897658	1.8748757	3.0162295	1.9204991	4.7565241	7.2005747	3.4768956	4.411863	2.6824355	4.6028764
	平均の下側95%	5.2769008	0.4584577	1.3171038	0.5795009	2.0768093	4.6327587	0.689771	1.4214704	0.4842311	1.5637903
	N	12	12	12	12	12	12	12	12	12	12
Shapiro-WilkのW検定	W	0.964067	0.808451	0.914289	0.779151	0.901178	0.838486	0.802427	0.935625	0.854661	0.886666
	p値(Prob<W)	0.8399	0.0117	0.2421	0.0055	0.1643	0.0265	0.01	0.4436	0.0419	0.1067
	正規分布適合の検定結果	正規分布	棄却	正規分布	棄却	正規分布	棄却	棄却	正規分布	棄却	正規分布
グラフ											

(2) t検定(分散が等しいと仮定)の結果

	総欠陥数	手戻り工数大	ビジネスニーズを満たせない	テストで検出できない	仕様書に書いていない	
t検定 分散が等しいと仮定	差	1.1667	-0.9167	-0.75	-0.3333	0.3333
	差の標準誤差	1.0069	0.7102	0.7813	0.585	0.9204
	差の上側信頼限界	3.2549	0.5562	0.8704	0.8798	2.2422
	差の下側信頼限界	-0.9216	-2.3896	-2.3704	-1.5465	-1.5756
	信頼率	0.95	0.95	0.95	0.95	0.95
	t値	1.158648	-1.29067	-0.9599	-0.56984	0.362143
	自由度	22	22	22	22	22
	p値(Prob> t)	0.259	0.2102	0.3475	0.5746	0.7207
	p値(Prob>t)	0.1295	0.8949	0.8262	0.7127	0.3604
	p値(Prob<t)	0.8705	0.1051	0.1738	0.2873	0.6396
グラフ						

※ t検定の結果、二群間に有意差は見られなかった。