

Revert に着目した不確かさに関する実証的分析

村岡 北斗^{1,a)} 鷗林 尚靖^{1,b)} 亀井 靖高^{1,c)} 佐藤 亮介^{1,d)}

概要：開発者の知識不足などの理由でソフトウェア開発中に発生する問題は、「不確かさ」と呼ばれている。本論文では、2つの調査課題について不確かさを分析した：RQ1) Revert に不確かさはどのように関わっているか、RQ2) 実際の不確かさは具体的にどのような問題の集合であるのか。結果として、RQ1 については、revert コミットではバグの原因が不確かであるため、その対応として revert を行ったものが多く存在し、reverted コミットでは例外対応したコミットが後に revert されるケースが多いことがわかった。一方、RQ2 については、OSS プロジェクトに存在する不確かさを、例外的な内容の不確かさ、バグと関連した不確かさ、将来的な変更についての不確かさ、変更内容に対する不確かさ、コードの内容に関わる不確かさの5つに分類することができた。また、最も多い不確かさはバグについての不確かさであることが分かった。解決すべき優先度が高い不確かさとして、バグの原因が曖昧な不確かさと例外的に不適切に処理された不確かさを挙げるができる。

キーワード：不確かさ、OSS、Revert コミット、Git、実証的分析

1. はじめに

ソフトウェア開発中の「不確かさ」と呼ばれる問題は、開発工程の様々な所に存在する。不確かさの例として、はっきりとしない要求、曖昧な仕様や、不明な API 仕様、修正内容が正しいか分からないなどといった問題が挙げられる。不確かさは、どのように扱っていいか一般的な対応方法がないため、開発者にとって扱いにくいもので一時的な措置をとられやすい。その結果として、不確かさはバグやソースコードの煩雑化の原因となりがちである。開発者は、不確かさを上手く管理したソフトウェア開発を行うことが必要とされている。近年、不確かさを抱擁したソフトウェア開発は重要な研究課題として、ソフトウェア工学において注目されている項目の1つである [4]。モデリング [2,3]、テスト・検証 [1,7]、開発環境 [13] など、ソフトウェア開発の様々なテーマのもとでの不確かさを扱った研究が存在する。その一方、不確かさはソフトウェア開発の各工程に存在している上に、対象となる問題の範囲が広すぎるため全体を捉えにくいという課題がある。そこで、本論文では2種類のアプローチをとることで不確かさを分析する。1つはソフトウェア開発に大きな悪影響を与える不確かさに対

象を絞って調査する方法である。対応すべき優先度が高い不確かさに対象を限定することで、効率的な不確かさの分析ができるのではないかと予想できる。もう1つは不確かさ全体をその内容によって分割する方法である。どのような内容の不確かさがどの程度存在するのか調査することによって、不確かさといった問題を小さく分割することができ、扱いやすくなるのではないかと考えられる。

本論文では、ソフトウェア開発に大きな悪影響を与える不確かさの1つとして、revert に関係した不確かさに注目した。Revert とは、版管理システム Git の機能の1つであり、過去に行った特定の更新内容を取り消してしまうものである。Revert が行われたコミットでは、不確かさが原因で必要なかったはずの作業が発生してしまったものではないかと仮定し、ソフトウェア開発に大きな悪影響を与えている不確かさとして分析の対象とした。本論文では、revert を行ったコミットを revert コミット、revert によって取り消されてしまったコミットを reverted コミットと表現する。Revert コミットに注目し、実際のソフトウェア開発プロジェクトにおいて、どのような不確かさが存在しているかを明らかにする。Revert コミットには不確かさがどの程度の割合存在し、どのような不確かさが存在するのか目視調査で分類を行うことで、不確かさについて分析する。本研究の位置付けと貢献

現状では不確かさに関する実証研究事例は極めて少なく、筆者らの既存研究 [11,12] が存在する程度である。我々

¹ 九州大学
Kyushu University
a) muraoka@posl.ait.kyushu-u.ac.jp
b) ubayashi@ait.kyushu-u.ac.jp
c) kamei@ait.kyushu-u.ac.jp
d) sato@ait.kyushu-u.ac.jp

は不確かさを分析するにあたって、コードの更新時に開発者がその更新内容について記述するコミットメッセージの情報を利用するアプローチをとっている。開発者はコードの変更内容についてコミットを行う際、不確かさな懸念を持っているときにその内容についてコミットメッセージに記述すると考えられるからである。そのため、我々の既存研究では Git で更新履歴が情報として公開されている OSS リポジトリをデータセットとして利用した。我々の既存研究ではコミットメッセージをテキストマイニングすることにより、ソフトウェア開発時にどのような不確かさが発生するかを明らかにした。その一方で、ソフトウェア開発に大きな悪影響を与える不確かさの分析については未解決のままであった。もし悪影響のある不確かさの傾向が分かれば、設計やコードのレビュー時に包容可能な不確かさか否かの判断指針とすることが可能となる。本論文では、revert という側面からこの問題にアプローチする。

本論文では、Git で管理されている 20 の OSS プロジェクトの開発履歴を対象に、以下の 2 つのリサーチクエスチョン (RQ) に答える。

RQ1: Revert に不確かさはどのように関わっているか

ソフトウェア開発に大きな悪影響を与える不確かさについて知るため、revert 機能に関する不確かさ候補コミットの内容について目視調査を行う。まず、Perez-Palacin ら [9] が提唱しているソフトウェア工学における不確かさの定義に基づいて、revert 機能に関する不確かさ候補コミットがどのようなものか調査する。次に、revert 理由に関する分類 [10] を用いて、revert が行われた理由と不確かさの関係について調査する。

RQ2: 不確かさはどのような内容で構成されているのか

不確かさといった問題が具体的にどのような問題の集合であるのかを把握するために、不確かさの内容について目視分類を行う。具体的には、開発者が開発中のどのような出来事について不確かであるのか、つまり何について開発者の知識が足りない状態であるかといった観点で分類を行った。実際のソフトウェア開発においてどのような不確かさに開発者が困っているか内容を分類することで、どのような不確かさが存在していて、どの不確かさが重要な問題なのか知ることができ、不確かさをうまく管理したソフトウェア開発について考えることができる。

以降、2 節では不確かさに関する関連研究を紹介し、本研究のアプローチを説明する。3 節では、不確かさの特定方法や目視調査の手法について説明し、4 節と 5 節で 2 つの RQ に対する調査の結果について述べる。6 節では、今回の分析内容について議論を行い、7 節にてまとめとする。

2. ソフトウェア開発での不確かさ

本節では、不確かさの分類に関わる関連研究を説明し、不確かさがどのように定義されているかを明らかにする。

表 1 不確かさを含むコミットメッセージの例

No.	コミットメッセージ
1.	Also remove some junk that was there for <u>unknown</u> reasons, this tool has a long history.
2.	This needs to be done in the makefile.msc too, but I'm <u>unsure</u> on the right way to do that.
3.	The file builds now and I only see warnings about using <u>unknown</u> Carbon API.
4.	It is highly <u>unclear</u> when to return FALSE.
5.	Whether or not undo memory should be included here is <u>debatable</u> .

2.1 ソフトウェア開発における不確かさ

ソフトウェア開発における不確かさは、要求分析、設計、実装方法やテストなどソフトウェア開発の様々な工程に存在している。はっきりと定まらない要求や、定まらない設計候補、未決定の実装手段など多種多様な内容の問題が不確かさとされている。こうした不確かさは、Perez-Palacin らの研究 [9] では、確率的な現象によって発生する不確かさと、開発プロセスに関する知識の不足を表す不確かさが存在すると定義されている。

OSS である GIMP プロジェクトで発見された実際の不確かさについて記述されているコミットメッセージの一部を表 1 に示す。1 つめの例について説明すると、なぜここに存在しているかわからなくなってしまった関数を削除してしまったといった内容で、開発者のプロジェクトのコードに関する知識が不足している不確かさである。2 つめの例では、正しい実装手段がわからないと言っていて、開発者が正しい実装を行えるだけの知識が不足しているといった不確かさである。

不確かさとはこのように様々な内容が存在しているため、全ての不確かさを一概に同様にして扱うことは難しい。そのため本論文では、実際の不確かさにはどのようなものがどのくらいの割合で存在しているかを調査することで、ソフトウェア開発における不確かさといった問題を扱いやすい問題に分割することを目的の 1 つとしている。

2.2 既存研究での不確かさの分類

2.2.1 ソフトウェア工学における不確かさの 3 つの分類

ソフトウェア開発における不確かさは大きく 3 つに分類される (*Known Knowns*, *Known Unknowns*, *Unknown Unknowns*) [1]。 *Known Knowns* は不確かさが存在しない開発である。 *Known Unknowns* はソフトウェア開発のプロセスの中で不確かな問題が存在する開発で、開発者には何が不確かであるか認識されている状態である。 *Unknown Unknowns* は開発者が何が不確かであるか問題が理解できていない状態での開発を指している。 *Unknown Unknowns* は開発者にも問題が理解できていない状態の不確かさのため、扱うことが難しい。そのため、本論文で扱う不確かさは、 *Known Unknowns* に該当するものを対象とする。

表 2 Perez-Palacin らによる不確かさの分類

観点	性質	性質の説明
場所	コンテキスト	環境に関する不確かさ
	モデル構造	モデル自体の構造に現れる不確かさ
	入力パラメータ	モデルへの入力に関する不確かさ
レベル	レベル 0	確定している知識
	レベル 1	知識の不足を認知している状態. 既知の不確かさ.
	レベル 2	知識の不足を認知できていない状態. 未知の不確かさ.
	レベル 3	不確かさを認知するプロセス自体が不足している状態.
	レベル 4	メタな不確かさ. 不確かさのレベル自体が不確か.
性質	認識的	十分なデータや知識が無いために発生する不確かさ
	偶発的	物理現象等の確率的な不確かさ

2.2.2 Perez-Palacin らによる不確かさの分類

Perez-Palacin らは、ソフトウェア工学における不確かさを場所、レベル、性質の3つの要素で分類する方法を提案している [9]. Perez-Palacin らが提案した不確かさの分類を表 2 に示す。不確かさの場所とは、不確かさがソフトウェアのどこに現れたのかを指している。不確かさのレベルは、どの程度開発者が問題に関する知識を持っているのかを段階的に分類したものである。レベル 0 は不確かではなく、レベル 1 以降、どの程度問題について開発者の知識が欠落している状態であるかに応じて、レベルが上がっていく。レベル 2 以上の不確かさでは、開発者が何がわかっていないのか把握できていない状態で、2.2.1 節で説明した *Unknown Unknowns* に相当する。また、不確かさの性質は不確かさが発生した状況についての分類である。認識的な不確かさとは、データの不確かさや、開発者の知識が足りないために発生する不確かさである。偶発的な不確かさは偶然性によってもたらされる不確かさである。

2.3 本論文のアプローチ

不確かさを管理したソフトウェア開発手法の提案をするために、実際のソフトウェア開発において存在する不確かさについて明らかにすることを目的とした調査を行う。本論文では、OSS リポジトリを対象に実際に存在している不確かさがどの程度存在しているのか、また、どのようなものが存在しているのかについて既存研究での不確かさ分類とオープンコーディング [5] を用いた不確かさ内容の目視分類の2通りの手法によって分類を行う。オープンコーディングとは対象のテキスト文書を簡単な文章で説明したものに変換し、似た内容のものについて組み合わせることで分類を行う手法である。

さらに、本論文では revert に関する不確かさに着目して実証研究を行う。revert というのは、版管理システム Git の機能の1つで、過去のコミットの内容を打ち消すコミットをする機能である。revert の原因の1つとして、要求や仕様、実装手段などに不確かな内容があることで、後に変

表 3 データセット概要

プロジェクト名	期間	全コミット数
apache/ant	2000/1/13-2017/12/12	13,710
apache/cassandra	2009/3/2-2017/11/21	23,687
apache/commons-codec	2003/4/25-2017/11/9	1,718
apache/commons-io	2002/1/25-2017/12/5	2,058
apache/commons-lang	2002/7/19-2017/12/20	5,128
apache/commons-logging	2001/8/2-2017/9/27	927
apache/derby	2004/8/11-2017/12/21	8,158
apache/james-mime4j	2005/5/3-2017/6/9	794
apache/tomcat	2006/3/27-2017/12/19	19,041
google/guava	2009/6/18-2017/12/20	4,520
mongodb/mongo	2007/10/19-2017/12/20	40,153
square/Aardvark	2014/10/5-2017/10/4	361
nicolasgramlich/AndEngine	2010/3/8-2013/12/11	18,005
android/frameworks-base	2008/10/21-2018/1/6	343,392
android/frameworks-opt-telephony	2012/6/13-2018/1/5	11,659
android/frameworks-support	2008/10/21-2017/12/7	35,641
android/tools-base	2012/12/10-2015/10/20	19,318
android/sdk	2008/10/21-2017/12/12	13,671
jedit/jedit-jEdit.bak	1998/9/27-2002/4/12	1,237
eclipse/eclipse.jdt.core	2001/6/5-2018/1/5	23,113
合計	—/—/—	570,086

更されたり、必要がなくなるなどの理由で、過去の編集内容を打ち消す必要が発生するといったケースが存在する。そのため、revert と不確かさは関係性が強いのではないかと推測される。

3. 調査手法

本節では、まず、不確かさをもつ revert コミットを特定する際に用いた、データセットとアプローチについて説明する。次に、特定したコミットに対して行った目視調査の手法について述べる。

3.1 データセット

本論文では Git で管理されている 20 の OSS リポジトリを対象とした。対象とした OSS のデータを表 3 に示す。これらのリポジトリはコードの不吉な匂いに関する実証研究 [8] でもデータセットとして扱われていたものである。

今回、OSS リポジトリの情報のうちコミットメッセージを不確かさが存在するかどうか判断するために利用した。開発履歴中の更新内容について記述するコミットメッセージには、更新内容に加えて、発生しているバグや不確かさなどの問題についても記述されている。そのため、データセットとして Git で管理されているプロジェクトである必要がある。また、今回の分析対象である revert コミットが存在するためには、ある程度の開発期間が必要だと予測される。これらの OSS は、開発期間も最短でも3年以上あり、コミット数も最小のプロジェクトで361件、最大で343,392件存在しているため、調査に十分なサイズのプロジェクトである。加えて、様々な種類の OSS を用意したものであるため多くの種類の不確かさが発見されることが期待できる。

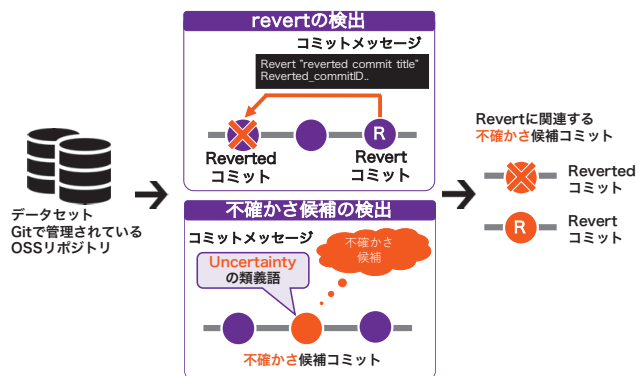


図 1 Revert に関連する不確かさ候補コミット特定手法

3.2 Revert に関連する不確かさ候補コミットの特定方法

Revert に関連する不確かさコミットを特定し分析する過程を図 1 に表す。

3.2.1 Revert に関連するコミットの特定

Revert コミットは、コミットメッセージの一部が定型文になっていて、revert されたコミットのコミットメッセージの 1 行目とコミット ID についての情報が書かれている。これを利用して、データセットの中から、revert コミットと revert によって取り消されたコミット（以下、reverted コミット）のペアをコミットコメントに正規表現を適用することで検出した。

3.2.2 不確かさの特定

開発者はコードの変更内容についてコミットを行う際、不確かさな懸念を持っているときにその内容についてコミットメッセージに記述すると考えられる。そこで本論文では、コミットメッセージ内に不確かさのキーワードを含んでいるコミットを不確かさ候補コミットと定義する。本論文では不確かさを表すキーワードは、Uncertainty の類義語を *Oxford American writer's thesaurus* [6] というシソーラス書籍から参照したものをを用いた。

3.2.3 調査対象コミットの定義

本稿では、不確かさ候補コミットのうち、revert コミットか reverted コミットであるものを対象とする。この 2 種類のコミットに分けて対象とする理由は、revert コミットと reverted コミットでは不確かさが revert 機能に対してどう関わったかが異なると予想されるためである。Revert コミットに不確かさが存在する場合は、後から不確かさに関する問題が発覚したもの、reverted コミットに不確かさが存在する場合は、不確かさへの対応が適切でなかったために revert が発生したものではないかと推測できる。

3.3 分析対象コミットの検出結果

Revert に関連するコミットの特定と不確かさ候補コミットをデータセットから抽出した結果を表 4 に示す。表中の割合は、プロジェクト全体のコミット数に対する不確かさを持つコミット数、revert コミット数の割合である。

全てのコミットの中に不確かさを含むコミットは 5,792 件（全てのコミットの 1.02%）、revert コミットは 3,214 件（全てのコミットの 0.56%）存在した。Revert コミットのうち、revert 時にコミットメッセージに自動で記入される reverted コミットのメッセージの 1 行目部分にキーワードを含むものは、重複を除くために、不確かさを含む revert コミットにはカウントせず、不確かさを含む reverted コミットとして集計とした。また、複数の revert コミットによって打ち消された reverted コミットは、revert された回数に関わらず 1 件として扱った。集計した結果、不確かさを含む revert コミットは 23 件、reverted コミットは 63 件存在した。Reverted コミットで複数回 revert されたものは 5 件存在していて、5 件とも revert された回数は 2 回であった。今回対象とした OSS リポジトリ 20 個の合計では、不確かさを含む revert は、全ての revert コミットの 2.83% を占めていた。全てのコミットに含まれる不確かさコミットの割合 1.02% であるので、revert コミットには普段のコミットより高い割合で不確かさが含まれる結果となった。しかし、OSS プロジェクトごとに比較すると不確かさを含む revert、reverted コミットが 0 件のリポジトリが 14 個存在し、プロジェクト単位で比較すると revert コミットには高い割合で不確かさが存在するとは言えない OSS の方が多く存在した。

この不確かさ候補コミットを用いて、以降 2 つの RQ のための分析を行う。

RQ1 では、ソフトウェア開発に大きな悪影響を与える不確かさについて知るため、不確かさ候補コミットのうち、revert コミットか reverted コミットであるコミットを対象として分析を行う。ここでは、複数人での目視分類を行うため、不確かさ候補コミットかつ reverted コミットであるものは、発見された 63 件のうちランダムに選んだ 10 件を分析対象とした。なお、目視分類は共著者のうち修士課程の学生 1 名と教員 2 名で実施した。教員による目視分類の負荷を低減するため、63 件全部ではなく 10 件に限定した。

RQ2 では、不確かさといった問題が具体的にどのような問題の集合であるのかを把握するために、不確かさの内容について目視分類を行う。対象となるコミットは、不確かさ候補コミットかつ revert コミットであるもの、不確かさ候補コミットかつ reverted コミットであるもの、revert 機能に関わっていない不確かさ候補コミットの 3 種類のコミットである。ここでも目視分類を行うため、revert コミットと reverted コミットについてはデータセットに存在した全てのコミットを扱うが、revert 機能に関わっていない不確かさ候補コミットはランダムに選んだ 100 件を対象とした。RQ2 では共著者のうち修士課程の学生 1 名が主に目視分類を行い、教員がその妥当性をチェックするアプローチをとった。そのため、RQ1 よりも多い 100 件を目視分類の対象とした。

表 4 不確かさ件数

project	全コミット数	不確かさを含む コミット数	割合 (%)	revert コミット数	割合 (%)	不確かさを含む revert コミット数	不確かさを含む reverted コミット数
apache/ant	13,710	231	1.68	3	0.02	0	1
apache/cassandra	23,687	61	0.26	108	0.46	0	0
apache/commons-codec	1,718	16	0.93	0	0.00	0	0
apache/commons-io	2,058	22	1.07	2	0.10	0	0
apache/commons-lang	5,128	26	0.51	9	0.18	0	0
apache/commons-logging	927	15	1.62	0	0.00	0	0
apache/derby	8,158	247	3.03	0	0.00	0	0
apache/james-mime4j	794	7	0.88	1	0.13	0	0
apache/tomcat	19,041	284	1.49	0	0.00	0	0
google/guava	4,520	165	3.65	0	0.00	0	0
mongodb/mongo	40,153	318	0.79	856	2.13	2	6
square/Aardvark	361	1	0.28	1	0.28	0	0
nicolasgramlich/AndEngine	1,800	13	0.72	3	0.17	0	0
android/frameworks-base	343,392	3,587	1.04	1,777	0.52	20	50
android/frameworks-opt-telephony	11,659	165	1.42	68	0.58	0	1
android/frameworks-support	35,641	250	0.70	306	0.86	1	4
android/tools-base	19,318	138	0.71	35	0.18	0	0
android/sdk	13,671	166	1.21	23	0.17	0	1
jedit/jedit-jEdit.bak	1,237	10	0.81	0	0.00	0	0
eclipse/eclipse.jdt.core	23,113	70	0.30	22	0.10	0	0
合計	570,086	5,792	1.02	3,214	0.56	23	63

4. RQ1:Revert に不確かさはどのように関わっているか

4.1 動機

不確かさはソフトウェア開発の各工程に存在している上に、対象となる問題の範囲が広すぎるため捉えにくい問題である。この不確かさの中でも、どのような内容の不確かさが優先して解決されるべき問題であるかは知られていない。対応すべき優先度が高い不確かさとはどのようなものか知ることで、効率的な不確かさ研究ができるのではないかと予想できる。

今回、ソフトウェア開発に大きな悪影響を与える不確かさの1つとして、revertに関連する不確かさに注目した。Revertが行われたコミットでは、不確かさが原因で必要なかったはずの工程が発生してしまったのではないかと考えられる。優先して解決されるべき不確かさの一例として、revertが行われたコミットに存在する不確かさはどのようなもので、revertと不確かさがどのように関わっているのか目視調査で分類を行う。

4.2 アプローチ

3節で検出された不確かさを含む revert コミット 23 件すべて、reverted コミットは 63 件のうち 10 件を対象に調査を行った。各不確かさ候補コミットのコミットメッセージの内容を読み、内容について評価する。不確かさという問題自体が曖昧なものであるため、不確かさの評価を行う際、主観性による影響を小さくする必要があると考えられる。そこで、本論文では、3名を調査者として、それぞれ

個別に不確かさの評価を行った。

まず、対象の不確かさ候補コミットについて、調査者が「該当コミットが本当に不確かさに関連したものであるのか」を判断する際の確信度を-2~2までの整数値を用いて5段階評価で評価する。不確かさの確信度について、分類結果が2だとコミットには不確かさが含まれている、-2はコミットには不確かさが含まれていないとし、0は不確かさであるともないとも判断できないとして評価を行う。

次に、不確かさの確信度が1以上である（つまり、どちらかといえば不確かさである）と判断した revert コミットと reverted コミットを対象として、revertが行われた不確かさコミットがどのようなものであるか、そしてどのような理由で revert されたのかについて調査を行った。

Revertに関連する不確かさがどのような性質のものであるのかを示すために、2.2.2節で説明した、Perez-Palacinらが定義した不確かさの分類の3要素を用いて、不確かさ候補コミットの評価を行い、集計した。

- 不確かさの場所 – コンテキスト, モデル構造, 入力パラメータ
- 不確かさのレベル – レベル1~4の4段階
- 不確かさの性質 – 認識的, 偶発的

また、どのような理由で revert されたのかについて調べるため、Shimagakiらの研究 [10] で用いられていた revert 理由に関する分類を利用し各コミットを分類した。

4.3 結果

不確かさの確信度について評価を行った結果のうち、revert コミットに関するものを表5に示す。表中のキー

表 5 revert コミットに不確かさがある場合の不確かさの確信度

プロジェクト	キーワード	revert コミット ID	reverted コミット ID	A	B	C	平均	中央	偏差
android/frameworks-base	may	9deba6	8ac32e	2	2	1	1.67	2.00	0.67
android/frameworks-base	risky	28f81d	d224c8	2	2	2	2.00	2.00	0.00
android/frameworks-base	may&probably	0f8ffd	b999cc	2	1	2	1.67	2.00	0.67
android/frameworks-base	probably	c82e7d	0cac71	2	2	2	2.00	2.00	0.00
android/frameworks-base	unclear	d6b42a	39aa79	2	2	2	2.00	2.00	0.00
android/frameworks-base	may	48b908	0a4e11	2	2	1	1.67	2.00	0.67
android/frameworks-base	probably	9309c1	—	2	2	2	2.00	2.00	0.00
android/frameworks-base	risky	75c21c	3127c2	2	2	2	2.00	2.00	0.00
android/frameworks-base	might	753e9e	c83d3e	2	2	2	2.00	2.00	0.00
android/frameworks-base	probably	4da482	059f8f	2	2	2	2.00	2.00	0.00
android/frameworks-base	unknown	ced57d	fe37ac	2	2	2	2.00	2.00	0.00
android/frameworks-base	unknown	ecd302	fe37ac	2	2	2	2.00	2.00	0.00
android/frameworks-base	probably	0b52ed	41bb49	2	2	2	2.00	2.00	0.00
android/frameworks-base	unknown	8b4838	77fe44	-1	0	-2	-1.00	-1.00	2.00
android/frameworks-base	might	7980d1	001d51	2	2	2	2.00	2.00	0.00
android/frameworks-base	probably	c5d8f6	27cf4a	2	1	2	1.67	2.00	0.67
android/frameworks-base	unknown	16626a	da3050	2	1	2	1.67	2.00	0.67
android/frameworks-base	probably	9b7a15	a47a9a	-2	1	1	0.00	1.00	6.00
android/frameworks-base	may	813805	1f5760	2	2	2	2.00	2.00	0.00
android/frameworks-base	might	b5428e	151819	2	1	0	1.00	1.00	2.00
android/frameworks-support	might	c56708	e5acd9	2	0	2	1.33	2.00	2.67
mongodb/mongo	may	7b9fe4	712c4f	2	1	2	1.67	2.00	0.67
mongodb/mongo	may	a7be5e	e6f9ef	2	2	1	1.67	2.00	0.67

表 6 Reverted コミットに不確かさがある場合の不確かさの確信度

プロジェクト	キーワード	revert コミット ID	reverted コミット ID	A	B	C	平均	中央	分散	revert 理由
android/frameworks-base	unknown	2ec157	3861bf	2	1	2	1.67	2	0.67	Defect introduced
android/frameworks-base	unknown	17548b	80ba0a	2	1	1	1.33	1	0.67	Obsolete solution
android/frameworks-base	unknown	3c9c09	17548b	2	1	1	1.33	1	0.67	Obsolete solution
android/frameworks-base	may	e4e7cd	a8a200	2	2	1	1.67	2	0.67	Obsolete solution
android/frameworks-base	unknown	e4e667	260c69	2	1	1	1.33	1	0.67	Obsolete solution
android/frameworks-base	unknown	2200da	993203	2	1	2	1.67	2	0.67	Incomplete fix
android/frameworks-base	may	4c5223	a8a200	2	2	1	1.67	2	0.67	Obsolete solution
apache/ant	may	296086	02aedb	2	0	2	1.33	2	2.67	Obsolete solution
mongodb/mongo	may	63d021	716e0f	2	2	2	2.00	2	0.00	Obsolete solution
mongodb/mongo	ambiguous	5c3f83	2eea3f	2	0	2	1.33	2	2.67	Obsolete solution

ワードは、コミットメッセージに含まれていた uncertainty の類義語である。また、A~C の列は調査者 3 人の評価結果を表している。

不確かさ候補コミットかつ revert コミットであるもののうち 18 件が、3 人とも不確かさが存在すると判断したコミットであった。不確かさ候補 revert コミット 23 件のうち、不確かさでないと判断されたコミットは 2 件存在していた。1 件はコミットメッセージにエラーメッセージが貼り付けてあるコミットで、エラーメッセージ部分にある unknown が不確かさのキーワードとして含まれているものであり、不確かさ候補コミットだと判定されたコミットであるが、調査者 3 人全てが不確かさだとは言えないと判断した。もう 1 件は、更新した内容がサイトに更新される

日付が不確かだといった内容で、これをソフトウェア開発上の不確かさとするか、別の領域の不確かさとするかで意見が分かれていた。

不確かさの確信度について評価した結果のうち、reverted コミットに関するものを表 6 に示す。reverted コミットのコミットメッセージに不確かさが含まれる 10 件には、不確かさでないとされたコミットは存在しなかった。

以降、不確かさでないと判断された revert コミットは 2 件を除いた、revert コミット 21 件と reverted コミット 10 件を対象にどのような不確かさで、なぜ revert されているのかについて分析を行った結果について述べる。

Reverted コミットでは、コミットメッセージ内に不確かさ候補コミットを見つける際に設定したキーワードの

「Unknown」が用いられているコミットが多く、コミットの内容は、未知の入力や想定されていないイベントに対応するためのコミット内容が正しくなく、後に revert されたケースが多い傾向にあった。unknown を含んでいた reverted コミットメッセージの一部を表 7 に示す。

4.3.1 Revert された不確かさの性質

Perez-Palacin らが定義した不確かさの分類を用いて、revert が行われた不確かさコミットがどのようなものであるか調査した結果を各項目ごとに示す。

不確かさの場所

不確かさを含む revert コミットには入力パラメータに存在するものはなく、コンテキストかモデル構造のどちらかにしか不確かさが見られなかった。また、コンテキストとモデル構造で意見が分かれたコミットが 22 件中 15 件と多数存在していた。

不確かさを含む reverted コミットでは、10 件のコミットのうち調査者の意見が完全に一致するケースが 1 件のみであった。revert コミットでは見られなかった入力パラメータに存在する不確かさが多い傾向となった。

不確かさのレベル

不確かさであると判断されたコミット全てが、不確かさのレベルは 1 であると判定されていた。これは、コミットメッセージに不確かさのキーワードを含むものを目視調査の対象としたことが原因であると考えられる。対象コミットは、コミットに認識された内容が記述されているコミットであるため、認知されていない不確かさであるレベル 2 以上のものが存在しない結果になったと推測される。また、不確かさでない判断されたコミットも対象から外しているためレベル 0 のものも存在せず、今回のケースでは不確かさのレベルが全て 1 になったと考えられる。

不確かさの性質

不確かさの確信度が 1 人以上の調査者に 1 以上と分類された 32 件のコミットのうち、30 件のコミットが調査者全てに認識的な不確かさであると分類された。残り 2 件については、認識的と偶発的で調査者の意見が分かれた。

Revert に関連する不確かさコミットがどのようなものであるか調べた結果、revert コミットはモデル構造に不確かさが存在するものが多く、reverted コミットでは入力パラメータに不確かさが存在していることが多いことがわかった。また、どちらのコミットも、十分なデータや知識が無いために発生する不確かさが多く存在していた。

4.3.2 不確かさコミットの Revert 理由

Revert コミットの内容は、バグに関するものが多数を占めていた。表 5 の 23 件中 12 件がコミットメッセージ内に

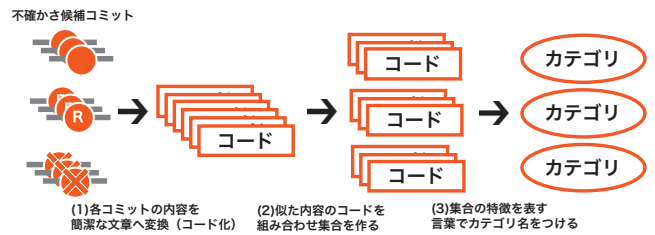


図 2 オープンコーディング概要

バグ ID を含んでいた。また、revert した理由ではバグの原因が曖昧だからとりあえずバグを発生させたコミットを取り消したケースが多く見られた。具体例を表 8 に示す。

不確かさコミットがどのような理由で revert されたか調査した結果、revert コミットではバグの原因が不確かであるため、その対応として revert を行ったものが多く存在し、reverted コミットでは例外対応したコミットが後に revert されるケースが多いことがわかった。

5. RQ2:不確かさはどのような内容で構成されているのか

5.1 動機

不確かさといった問題が具体的にどのような問題の集合でいて、どのような内容のものが存在しているかについて実際のソフトウェア開発に存在する不確かさについて調査されていない。実際のソフトウェア開発においてどのような不確かさに開発者が困っているか内容を分類することで、どのような不確かさが存在していて、どの不確かさが重要な問題なのか知ることができる。また不確かさといった大きな問題を切り分けることで、不確かさをうまく管理したソフトウェア開発についての議論が行いやすくなると思われる。そこで、本節では、不確かさといった問題が具体的にどのような問題の集合であるのかを把握するために、不確かさの内容について目視分類での調査を行う。

5.2 アプローチ

3.2 節で発見された revert に関連した不確かさ候補コミットについては全てを、revert に関わっていない不確かさ候補コミットはランダムに 100 件を選び対象とし、各不確かさ候補コミットについて目視分類を行った。今回、オープンコーディング [5] を用いて不確かさの内容について分類を行う。手法の概略図を図 2 に示す。まず、コミットコメントの内容を簡単な文章で説明したもの（コード）に変換し、似た内容のものについて組み合わせ、各集合に名前をつけることでカテゴリとした。また、この過程で不確かさ候補コミットに記述されている内容が不確かさかどうかの判定も行った。

目視分類を行う際、コミットメッセージに書かれている

表 7 Unknown を含む reverted コミット

プロジェクト	コミット ID	revert コミットメッセージの一部
android/frameworks-base	17548b	Revert "be more robust with handling unknown sensors"
android/frameworks-base	e4e667	Revert "Treat UNKNOWN tootype events as Mouse events."

表 8 バグと関連する revert コミット

プロジェクト	コミット ID	revert コミットメッセージの一部
android/frameworks-base	ecd302	Bug 6357880 has unknown causes at the moment but this change is the most likely candidate.
mongodb/mongo	7b9fe4	Unit test is failing on buildslaves; counts of records for indexes may be messing things up

内容では開発者が何に対して理解できていない状態なのかといった内容に注目して分類した。不確かさの分類には 2 節で説明した, Perez-Palacin らの研究 [9] が存在しているが, これは不確かさがどこに存在していて, どの程度理解できていないか, そしてどのような性質の不確かさなのかについて分類を行ったものであり, 今回の不確かさの事例に関する分類とは異なるものである。

5.3 結果

不確かさ候補コミットのうち revert コミット 23 件, reverted コミット 63 件, どちらでもないコミット 100 件についてコミットに含まれる不確かさについて目視分類を行った。分類された不確かさ候補コミットのカテゴリごとの件数を表 9 に示す。

コミットメッセージに現れる主要な不確かさを 5 種類のカテゴリに分類することができた。また, そのうち 4 つのカテゴリではサブカテゴリとしてさらに細かいカテゴリを作成することができた。実際に不確かさだと判断された不確かさ候補コミットは全体の 94.6%であった。本論文では, 分類手法としてまずコミットコメントをコード化し, 似た内容のものを組み合わせることでカテゴリを作成した。各カテゴリとサブカテゴリに分類されたコミットの具体例として, コミットコメントの一部を表 10 に示す。そこで, 各カテゴリごとにどのようなコミットをカテゴリにまとめたのか, 内容の説明と結果に関する考察を以下で行う。

例外処理と関連した不確かさ

このカテゴリには, 例外処理に関わっている不確かさが多く見られた。

サブカテゴリ「未知の入力」には入力が不確かなコミットが分類されている。このサブカテゴリのコミットの内容は未知の入力への対応を行なっている, いわゆる例外処理を追加したといったものが多く見られた。また, 不確かさを見つける際に用いたキーワードが reverted コミットの 7 件全て, revert でないコミットでは 13 件中 12 件が unknown であるといった特徴がみられた。

サブカテゴリ「出力結果が曖昧」には, ある入力によってどのような振る舞いをとるのが不確かだとしているコミットを分類した。このサブカテゴリのコミットの内容には, あるパターンの入力に対して例外処理ができていない

ために, 振る舞いがどのようになるかが曖昧であるといった内容のものが存在していた。

このカテゴリの特徴としては revert コミットにはこの内容の不確かさは存在していないが, reverted コミットには多く見られる内容の不確かさであることが言える。例外処理について書かれている不確かさは, 後に revert されてしまう可能性が高い重要な不確かさだと考えられる。

バグと関連した不確かさ

バグに関係した内容の不確かさが分類されているカテゴリである。バグの原因, 影響, 修正のどの部分が不確かなのかで 3 つのサブカテゴリに分かれている。「バグの原因が不確か」に含まれているコミットは, なぜバグが発生しているのかわからないといった内容のコミットが含まれている。「バグの影響が不確か」に含まれているコミットでは, バグによって発生した不具合の内容や規模が不確かだといったコミットが含まれている。「バグの修正が不確か」では, バグの修正ができたかもしれない, できていないかもしれないといった内容のコミットが多く分類された。

今回使用したデータセットでは, revert に関係しているコミット, 関係していないコミットどちらにおいても, 一番不確かさが多く分類されたカテゴリとなった。また, サブカテゴリでは「バグの原因が不確か」な不確かさが多い結果になった。このカテゴリが含まれる不確かさが, 開発者を悩ませている頻度が最も高い内容のものだと言える。

将来的な変更についての不確かさ

将来的に改善や修正のために違うアプローチによる実装が必要とされるかもしれないという内容の不確かさがこのカテゴリには含まれている。将来的に問題が出るかもしれないから, その際修正が必要かもしれないといった内容のものがサブカテゴリの「修正が必要になるリスク」に含まれている不確かさである。また, もうひとつのサブカテゴリの「代替案の提示」には, 他の実装方法が優れているかもしれないといった内容のコミットが分類されている。

例外的な内容の不確かさと同様に, revert コミットに対し reverted コミットが多く存在し, 後に revert されてしまう可能性が高い重要な不確かさではないかと考えられる。

変更内容に対する不確かさ

そのコミットで行った変更内容が不十分であるかもしれ

表 9 不確かさ分類結果

カテゴリ	サブカテゴリ	revert コミット	reverted コミット	revert なしコミット
例外的な内容の不確かさ	未知の入力	0	7	13
	出力結果が曖昧	0	2	2
バグと関連した不確かさ	バグの原因が不確か	11	16	29
	バグの影響が不確か	2	7	10
	バグの修正が不確か	2	5	6
将来的な変更についての不確かさ	修正が必要になるリスク	0	1	3
	代替案の提示	1	4	14
変更内容についての不確かさ	十分でない可能性	3	5	4
	不具合追加の可能性	0	5	2
コードの内容に関する不確かさ		0	2	4
その他の不確かさ		3	4	9
不確かさではない		1	5	4
合計		23	63	100

表 10 各カテゴリの具体例

カテゴリ	サブカテゴリ	該当コミットのコミットメッセージの一部
例外的な内容の不確かさ	未知の入力	be more robust with handling unknown sensors ...
	出力結果が曖昧	... Reject ambiguous positional projections and updates
バグと関連した不確かさ	バグの原因が不確か	Bug 6357880 has unknown causes at the moment but this change is the most likely candidate.
	バグの影響が不確か	The change broke SetupWizard and probably other apps
	バグの修正が不確か	... Add logging for assigning home stack to non-home task to track down bug. And maybe fix bug ...
将来的な変更についての不確かさ	修正が必要になるリスク	... which is additional complexity and risk .
	代替案の提示	... It might be useful for processing touch events over "desktop" in freeform MW mode.
変更内容についての不確かさ	十分でない可能性	...The existing spec is unclear ... , need more clarification.
	不具合追加の可能性	... This may break some apps, we'll see how it goes. If it is too bad ...
コードの内容に関する不確かさ		Note the interpolated values may differ slightly from the framework ...

ない、新たな問題を生み出してしまったかもしれないといった内容の不確かさが含まれているカテゴリである。サブカテゴリの「十分ではない可能性」では、実現しなかった機能を今の更新では満たしていないのではないかとといった内容の不確かさが分類されている。「不具合追加の可能性」では、そのコミットで新たにバグを生み出してしまったかもしれないといったコミットコメントが多く存在した。

コードの内容に関わる不確かさ

コミッタがプログラムの振る舞いが理解できていないため、変更が正しいかどうか判断ができない不確かさが分類されている。

不確かさは5つのカテゴリに分類することができた。カテゴリ別でみると「バグに関連した内容の不確かさ」が、47.3%と他のカテゴリと比べても多く存在し、開発者を悩ませる頻度をもっとも高い内容の不確かさではないかと考えられる。また、カテゴリ「例外的な内容の不確かさ」と「将来的な変更についての不確かさ」は、revert コミットに対し reverted コミットの数が多く、後に revert されてしまう危険性の高い不確かさではないかと考えられる。

6. 妥当性への脅威

6.1 構成概念妥当性への脅威

RQ1 では、ソフトウェア開発に大きな悪影響を与える不確かさの1つとして、不確かさがどのように revert に関わっているのかを調査した。今回の調査では、revert コミットと reverted コミットを対象とし、実際に発見された不確かさについて調べることでソフトウェア開発に大きな悪影響を与える不確かさの傾向とした。RQ1 の結論として、revert コミットではバグの原因が不確かであるため、その対応として revert を行ったものが多く、reverted コミットでは例外対応したコミットが後に revert されるケースが多いことが分かった。しかし一方、Shimagaki らの研究 [10] でも、バグの混入や一時的な対応などが revert の要因となっていることが判明している。今回の分析対象が OSS であるのに対し、[10] は企業開発ソフトウェアを主な分析対象としているという違いがあるものの、両者に類似の傾向があるのは興味深い。[10] は不確かさを対象にしているものの、revert の背景として不確かさが関連している可能性はあり、その究明は今後の研究課題である。

6.2 外的妥当性への脅威

今回の実証分析では、20個のOSSプロジェクトを対象に分析対象とした。しかし、実際に目視調査を行った不確かさを含む revert コミット, reverted コミットは4つのOSSリポジトリにしか存在していなかった。また、OSSプロジェクトごとのコミット数をみても android/frameworks-base のコミット数が非常に多く、コミット数で見ると全体の約6割を占めている。さらに、4節で目視調査を行ったコミット33件のうち、28件が android/frameworks-base プロジェクトのものである。今回、revertに関連する不確かさについて見つかった傾向が一般的なものでなく、android/frameworks-base プロジェクト固有の傾向である可能性が存在する。今後、データセットを増やすことで、さらに一般化した不確かさについて確認したい。

不確かさの調査を行う際、不確かさを分類するための情報が十分でない可能性があるといった問題が存在する。実証分析として、OSSの開発履歴から不確かさについて判断するが、開発履歴には開発時に開発者が考えていたことが全て記述されているとは限らない。今回、分析にはコミットメッセージを用いたが、コミットメッセージから開発者が考えていることが一意に読み取れないケースも存在した。また、当然開発者が気づいていない問題についても記述されていない。つまり、2.2.1節で説明した *Unknown Unknown* に値する不確かさについては、特定ができていない。4節の目視調査結果も、不確かさのレベルは全てのコミットがレベル1、つまり、知識の不足を認識している状態、*Known Unknowns* の不確かさしか見つけることができなかった。

6.3 内的妥当性への脅威

RQ2の目視分類については、分析者の主観による影響を考慮する必要がある。しかしながら、分析結果は我々のソフトウェア開発経験と大きく乖離した内容ではなく、むしろ経験を裏付ける内容となっている。主観がある程度入り込むのは避けられないが、今回の分析については妥当な結果が得られたと考えられる。

7. まとめ

本論文では、20個のOSSリポジトリのコミットメッセージを分析対象として2つのRQを立て、実際のプロジェクトで発生する不確かさについて調査した。

本論文の結果をもとに考えられる今後の研究課題として、出現頻度が高い「バグに関連した内容の不確かさ」や、revertされる危険性が高い「例外的な内容の不確かさ」や「将来的な変更についての不確かさ」について分析し、これらの不確かさに対応した開発手法について提案していくといったことが考えられる。

謝辞

本研究は JSPS 科研費 JP26240007 の助成を受けた。

参考文献

- [1] Elbaum, S. and Rosenblum, D. S.: Known Unknowns: Testing in the Presence of Uncertainty, *Proceedings of the 22nd International Symposium on Foundations of Software Engineering*, pp. 833–836 (2014).
- [2] Esfahani, N., Razavi, K. and Malek, S.: Dealing with Uncertainty in Early Software Architecture, *Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering*, p. 21 (2012).
- [3] Famelis, M., Salay, R. and Chechik, M.: Partial Models: Towards Modeling and Reasoning with Uncertainty, *Proceedings of the 34th International Conference on Software Engineering*, pp. 573–583 (2012).
- [4] Garlan, D.: Software Engineering in an Uncertain World, *Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research*, ACM, pp. 125–128 (2010).
- [5] Khandkar, S. H.: Open Coding, *University of Calgary*, Vol. 23, p. 2009 (2009).
- [6] Lindberg, C. A.(ed.): *Oxford American Writer's Thesaurus*, Oxford University Press (2012).
- [7] Llerena, Y. R. S.: Dealing with Uncertainty in Verification of Nondeterministic Systems, *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, pp. 787–790 (2014).
- [8] Palomba, F., Nucci, D. D., Tufano, M., Bavota, G., Oliveto, R., Poshyvanyk, D. and De Lucia, A.: Landfill: An Open Dataset of Code Smells with Public Evaluation, *Proceedings of the 12th Working Conference on Mining Software Repositories*, IEEE Press, pp. 482–485 (2015).
- [9] Perez-Palacin, D. and Mirandola, R.: Uncertainties in the Modeling of Self-adaptive Systems: A Taxonomy and an Example of Availability Evaluation, *Proceedings of the 5th ACM/SPEC International Conference on Performance Engineering*, pp. 3–14 (2014).
- [10] Shimagaki, J., Kamei, Y., McIntosh, S., Pursehouse, D. and Ubayashi, N.: Why are Commits being Reverted?: A Comparative Study of Industrial and Open Source Projects, *Proceedings of the Software Maintenance and Evolution (ICSME), 2016 IEEE International Conference on*, IEEE, pp. 301–311 (2016).
- [11] Ubayashi, N., Kamei, Y. and Sato, R.: When and Why Do Software Developers Face Uncertainty?, *Proceedings of the 2019 IEEE International Conference on Software Quality, Reliability and Security*, to appear (2019).
- [12] Ubayashi, N., Muraoka, H., Muramoto, D., Kamei, Y. and Sato, R.: Exploring Uncertainty in GitHub OSS Projects—When and How Do Developers Face Uncertainty?—, *Proceedings of the 40th International Conference on Software Engineering (Companion Volume: Poster)*, pp. 272–273 (2018).
- [13] Ubayashi, N., Takuya, W., Kamei, Y. and Sato, R.: Git-based Integrated Uncertainty Manager, *Proceedings of the 41st International Conference on Software Engineering (Companion Volume: Demonstrations)*, pp. 95–98 (2019).