

要求仕様の漏れ・抜け検証の手法について

株式会社 NTT データ MSE
キャッツ株式会社

松井 恭

1. はじめに

ソフトウェア開発において、要求仕様の精度向上の重要性は多くの人が認める所ではありますが、この要求仕様書の作成は極めて属人的であり、その品質が起因する後工程での手戻り等が頻発していることも否定できないのが現実ではないでしょうか。

STD、STM、TRD、SC 等、設計手法にはその開発特性や用途に応じて様々存在しますが、仕様の漏れ・抜けには事象と状態の組み合わせのすべてを網羅的に考慮できる STM が最も有効です。しかしながらこの STM 設計手法を実際の開発現場で運用されている組織にお聞きすると、以下の様な点に苦心されているというお話を耳にします。

(1)STM まで展開できれば非常に効果的であるが、状態とイベントの抽出は属人的になっている。

(2)STM への展開スキルを持った人材の育成が難しい。

(3)仕様書を機械的に STM へ展開しても、仕様書に漏れがあったら見つけられない。

STM 開発手法は、“表”に落とし込めれば、「条件の記載がない箇所」、「処理、遷移先の記載がない箇所」については検出できますが、それ以前、即ち“表”の作成過程で検出される「状態・契機・条件などの記載がない箇所」「抽象度が高く細分化が必要な箇所」については属人的になり、“表”に落とし込めない項目(=見えない漏れ抜け)の発生確率は低減できません。

本寄稿では、NTT データ MSE が展開するサービスである『要求仕様の漏れ・抜け検証(Axis)』の手法の一部を利用し、STM 利用の有効性を更に高める方法についてご紹介します。

2. STMによる仕様の漏れ・抜け対策

まず始めに具体例として要求仕様を図1に示し、STMを用いた仕様の漏れ・抜け検証を行います。

要求番号	要求事項
1	タイマボタンを押すことで、時間を分で設定し、タイマを起動できる。
1-1	タイマが起動している／していないにかかわらず、タイマボタンを押す度にタイムアップまでの残り時間に1分加算し、タイム値に設定する。
1-2	設定したタイム値を分単位でタイマ残り時間表示窓に表示する。
1-3	タイマが起動していない場合は、タイム値を設定し終え、タイマボタンから手を離して1秒経過したらタイマが起動する。
1-4	蓋が閉じているときのみ、タイマボタンを有効にする。
2	タイマボタンを長く押すことで、タイマをリセットし、タイマを停止できる。
2-1	タイマが起動している場合、タイマボタンを長押ししたら、ブザーを1回鳴らした後、タイム値はリセットされ、タイマが停止する。
3	ユーザが指定した時間がきたら、ブザーを鳴らして知らせる。
3-1	タイムアップしたら、ブザーを3回鳴らす。

図1 タイマ機能付き電気ポットの要求仕様書

上記の要求仕様をSTMに展開すると図2の通りになります。

		タイマ停止中	タイマ起動中
タイマボタン 押下	短押し	蓋オープン 残り時間に1分加算し、 タイム値に設定 分単位で タイマ残り時間表示窓 に表示	蓋オープン 残り時間に1分加算し、 タイム値に設定 分単位で タイマ残り時間表示窓 に表示
	長押し		蓋オープン ブザーを1回鳴らす タイマ停止 設定した時間をリセット タイマ停止中
タイマボタンから 手を離して1秒経過	タイマ起動 タイマ起動		
タイムアップ			ブザーを3回鳴らす

「長押しした場合」の時は？
処理、遷移先は？

遷移先は？

図2 要求仕様書のSTMへの展開

図2に示す通り、「タイマ停止中にタイマボタンを長押しした際の処理や遷移先」「タイマ起動中にタイムアップした際の遷移先」の明確化が必要であることが分かります。

この結果を要求仕様にフィードバックすると要求仕様は図3に示す形となり、赤字箇所仕様が不足していた事が分かります。

要求番号	要求事項
1	タイマボタンを押すことで、時間を分で設定し、タイマを起動できる。
1-1	タイマが起動している／していないにかかわらず、タイマボタンを押す度にタイムアップまでの残り時間に1分加算し、タイム値に設定する。
1-2	設定したタイム値を分単位でタイマ残り時間表示窓に表示する。
1-3	タイマが起動していない場合は、タイム値を設定し終え、タイマボタンから手を離して1秒経過したらタイマを起動する。
1-4	蓋が開いている場合、タイマボタンを操作無効にする。
2	タイマボタンを長く押すことで、タイマをリセットし、タイマを停止できる。
2-1	タイマが起動している場合、タイマボタンを長押ししたら、ブザーを1回鳴らした後、タイム値はリセットされ、タイマが停止する。
2-2	タイマが起動していない場合は、ブザーを1回鳴らした後、 タイム値をリセットする。
3	ユーザが指定した時間がきたら、ブザーを鳴らして知らせる。
3-1	タイムアップしたら、ブザーを3回鳴らし、 タイマを停止した状態にする。

図3 要求仕様書 (STM 開発手法検出後)

この様に、通常のSTM開発手法では、「条件の記載がない箇所」、「処理、遷移先の記載がない箇所」を検出することが可能です。

3. 要求仕様の漏れ・抜け検証手法

「要求仕様の漏れ・抜け検証(Axis)」は、文書を係り受けや属性によって論理的に分解、解析する「網羅性検証」と、同値分割、限界値分析などのテスト技法を応用し具体性を向上させる「具体性検証」の大きく二つの段階が有りますが、今回は「網羅性検証」に焦点を当て、ご説明します。

網羅性検証では「要求仕様整理表」を用い、要求仕様書文書の仕様の意味のある文章ブロックに分け、各文章中の「～したら」、「～の場合」、「～の時に」、「～され」、「～する」等のキーワードに着目し、係り受けを分析しながら「契機」、

「遷移元状態」、「条件」、「処理」及び「遷移先状態」の5つの項目に整理して記載します。要求仕様整理表のフォーマットを図4に、具体的な記載方法を図5に示します。

文書番号	契機		遷移元状態		条件		処理		遷移先状態	
	対象	要素	対象	要素	対象	要素	対象	要素	対象	要素
	文書原文	イベント ター	イベント ムーバー	Before ムーバー	コンディ ション	コンディ ション	アクター	アク ション	After ムーバー	ステー タス
要求仕様整理表										

図4 要求仕様整理表

タイマが起動している場合、タイマボタンを長押ししたら、ブザーを1回鳴らした後、タイマ値はリセットされ、タイマが停止する。

文書番号	契機		遷移元状態		条件		処理		遷移先状態	
	対象	要素	対象	要素	対象	要素	対象	要素	対象	要素
2-1	文書原文	イベント ター	イベント ボタン	Before ムーバー	コンディ ション	コンディ ション	アクター	アク ション	After ムーバー	ステー タス
		タイマが起 動している 場合、タイ マが停止 する。	タイマ ボタン	タイマ	起動 中		(1) ブザー (2) タイマ 値		タイマ	停止 中

図5 要求仕様の分解

この手法で実際に図1の原文を要求仕様整理表へ展開した結果が図6となります。「契機」、「遷移元状態」、「条件」、「処理」及び「遷移先状態」に展開した結果、黄色で示した箇所の様に記述されていない部分が明確化されます。

文書番号	文書原文	契機		遷移元状態		条件		処理		遷移先状態	
		対象	要素	対象	要素	対象	要素	対象	要素	対象	要素
1-1	タイマが起動している/押し、タイマをリセットする。	タイマ	ボタン	タイマ	停止中			タイマ	タイマ	タイマ	タイマ
1-2	設定したタイマで経過分単位でブザーを鳴らす。タイマ値はリセットされる。	タイマ	ボタン	タイマ	停止中			タイマ	タイマ	タイマ	タイマ
1-3	タイマが起動している場合、タイマボタンを長押ししたら、ブザーを1回鳴らした後、タイマ値はリセットされ、タイマが停止する。	タイマ	ボタン	タイマ	起動中			タイマ	タイマ	タイマ	タイマ
1-4	タイマが起動している場合、タイマボタンを長押ししたら、ブザーを1回鳴らした後、タイマ値はリセットされ、タイマが停止する。	タイマ	ボタン	タイマ	起動中			タイマ	タイマ	タイマ	タイマ
2-1	タイマが起動している場合、タイマボタンを長押ししたら、ブザーを1回鳴らした後、タイマ値はリセットされ、タイマが停止する。	タイマ	ボタン	タイマ	起動中			タイマ	タイマ	タイマ	タイマ
2-2	タイマが起動している場合、タイマボタンを長押ししたら、ブザーを1回鳴らした後、タイマ値はリセットされ、タイマが停止する。	タイマ	ボタン	タイマ	起動中			タイマ	タイマ	タイマ	タイマ

図6 原文から要求仕様整理表への展開結果

この結果に対し「不明な状態を検討した結果”設定中”という状態が必要である」と決定したと仮定します。そうすると、この新たに出現した「設定中」という状態は要求仕様整理表中の遷移元の列にも埋め込む必要があることが明確化され、それに伴い、図7の水色で示した箇所の様に、更に明確化を必要とする箇所が出現してきます。

文書番号	文書原文	契機		遷移元状態		条件		処理		遷移先状態	
		対象	要素	対象	要素	対象	要素	対象	要素	対象	要素
1-1	タイマが起動している/押し、タイマをリセットする。	タイマ	ボタン	タイマ	停止中			タイマ	タイマ	タイマ	タイマ
1-2	設定したタイマで経過分単位でブザーを鳴らす。タイマ値はリセットされる。	タイマ	ボタン	タイマ	設定中			タイマ	タイマ	タイマ	タイマ
1-3	タイマが起動している場合、タイマボタンを長押ししたら、ブザーを1回鳴らした後、タイマ値はリセットされ、タイマが停止する。	タイマ	ボタン	タイマ	起動中			タイマ	タイマ	タイマ	タイマ

図7 検出した状態を展開

また、新たに状態が増えたことにより図7に示すピンク色箇所の様に既に定義済みの「遷移元状態」と「遷移先状態」の関係も再度検討する必要があります。

これを検討すると図8の様に、起動中への遷移は“停止中”からではなく、新たに追加した“設定中”からの遷移へ変更する必要がある事が分かります。

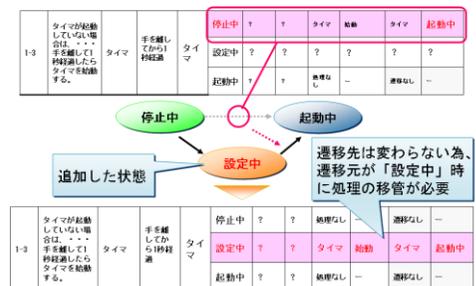


図8 遷移元/遷移先状態の再検討

同様に、条件に対しても図9に示す様に、同じ契機に対応する条件を元に不明な条件の検討をしていきます。

変更番号	変更事項	イベント	Before	コンディショナ	コンディショナ	アクター	アクション	After	
1-1	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
1-1	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	?	?	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			設定中	?	?	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	?	?	タイマ値	残り時間が0に達し、設定	タイマ	設定中

図9 検出した条件を展開

全ての状態に対して、それぞれに設定した条件を反映した要求仕様整理表を図10に示します。

変更番号	変更事項	イベント	Before	コンディショナ	コンディショナ	アクター	アクション	After	
1-1	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			停止中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中

図10 条件を展開した要求仕様整理表

全ての網羅性検証を行った後の要求仕様整理表は図11の通りとなり、黄色で示した部分が新たに追記された部分となります。

変更番号	変更事項	イベント	Before	コンディショナ	コンディショナ	アクター	アクション	After	
1-1	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
1-2	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
1-3	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
1-4	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
2-1	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
2-2	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
3-1	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中

図11 要求仕様整理表 (AXIS 検証後)

この結果は図12の様に容易にSTMへ展開が可能です。その結果が図13となり、図2と比較し、要件の精度が向上しています。

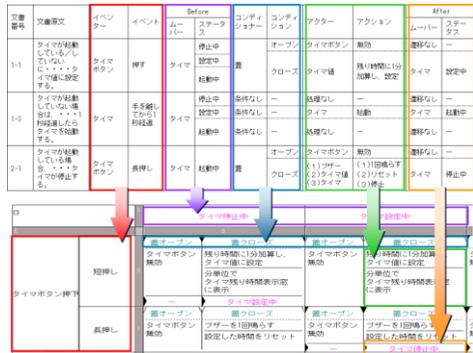


図12 要求仕様書からSTMへ展開

変更番号	変更事項	イベント	Before	コンディショナ	コンディショナ	アクター	アクション	After	
1-1	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
1-2	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
1-3	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
1-4	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
2-1	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
2-2	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
3-1	タイマが起動している間は、タイマがスタート時にタイマが設定されている値にタイマが加算し、タイマで検知する。	タイマ	停止中	差	オープン	タイマボタン	無効	遷移なし	—
			設定中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中
			起動中	差	クローズ	タイマ値	残り時間が0に達し、設定	タイマ	設定中

図13 網羅性検証後のSTM

また、この分析結果を要求仕様にフィードバックすると要求仕様は図14に示す形となり、図3と比べ、更に赤字箇所の仕様が不足していた事が分かります。

要求番号	要求事項
1	タイマボタンを押すことで、時間を分で設定し、タイマを起動できる。
1-1	タイマが起動している/していないにかかわらず、タイマボタンを押される際にタイマアップまでの残り時間に分加算し、タイマ値に設定する。タイマ値設定後から確立するまでの間は、タイマ設定中の状態にする。
1-2	すべての状態において、設定したタイマ値を分単位でタイマ残り時間表示窓に表示する。
1-3	タイマ設定中にタイマボタンから手を離して秒経過した時点でタイマ設定確立とし、タイマを始動する。
1-4	タイマが起動している/していない状態でタイマボタンから手を離して秒経過後もタイマの始動は行わない。
2	すべての状態において、蓋が開いている場合は、タイマボタンは無効とする。
2-1	タイマボタンを長く押すことで、タイマをリセットし、プザーを1回鳴らした後に、タイマ値はリセットされ、タイマが停止する。
2-2	タイマが起動している状態でタイマアップしたら、プザーを3回鳴らしたら、タイマを停止した状態とする。
3	タイマが起動している/いない/タイマ設定中にタイマアップが発生した場合は、何も処理を実行しない。

図14 AXIS 検証後の要求仕様書

4. 最後に

今回ご紹介した手法は、必ずしも STM のみに適応するものではなく、要求仕様書自体の検証サービスとして単独でも展開しており、漏れ・抜けの存在をユーザー様にご報告し、効果を上げています。

また要件トレースにおいて必須となる、粒度揃えの手段としての試行も開始しており、手法そのものの合理性と確実性に関しても大学関係にアドバイスを頂きながら向上を図っています。

しかしながら、手法構築の発端は STM 導入にかかわる問題解決であり、ZIPC を使った STM による設計開発をより精度の高いものに出れると考えておりますので、この様な手法とあわせ、ZIPC ユーザーの皆様様のソフトウェア開発効率向上の一助となれば幸いです。

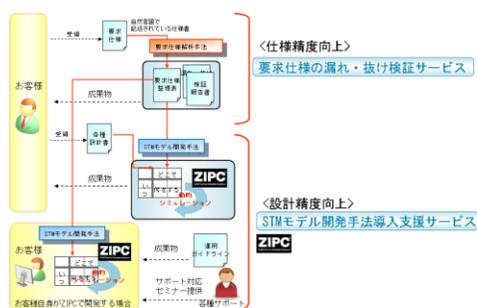


図 15 AXIS サービス