

データベースを利用したシステムにおける

新たな同値クラスのデータの発生にともなう影響分析手法の提案

主査 : 奥村 有紀子 (有限会社デバッグ工学研究所)
副主査 : 秋山 浩一 (富士ゼロックス株式会社)
: 喜多 義弘 (東京工科大学)
アドバイザー : 堀田 文明 (有限会社デバッグ工学研究所)
研究員 : 別役 雅洋 (東洋ビジネスエンジニアリング株式会社)

概要

データベースを利用したシステムにおいて、変更前には発生し得なかった、新たな同値クラスのデータが発生する修正を行うと、そのデータを参照している他機能にも影響が波及する。対象のシステムの仕様と設計内容の知識が浅い担当者が、影響分析を実施する場合、他機能の修正の必要性に気づくことができない。その原因は、テーブルの非正規化により同じ項目が複数のテーブルに定義されていることによって、影響分析の作業が複雑になることが原因であると仮定した。本研究では類似の先行研究を参考にして、新たな影響分析手法を考案した。考案した手法を設計レビューで適用した結果、修正すべき機能の漏れを検知することができた。ただし、考案した手法の適用件数が少ないため、有効性を判断することはできていない。今後、更なる研究を継続する必要がある。

1. はじめに

本研究は、データベース（以降、DBと呼ぶ）のテーブルを更新する機能の追加・修正が発生した場合の影響分析を対象とする。機能の追加・修正を総称してシステム変更と呼ぶ。

DBを利用したシステムでは、DBのテーブルから取得したデータをもとに、条件分岐が行われる。条件分岐に影響する単位でのデータの固まりを、本研究では、同値クラスのデータと呼ぶ。具体例を図1に示す。テーブルの列（以降、項目と呼ぶ）の値が、0以上の数か負の数かで分岐する処理が存在した場合、2つの同値クラスが存在するものとする。

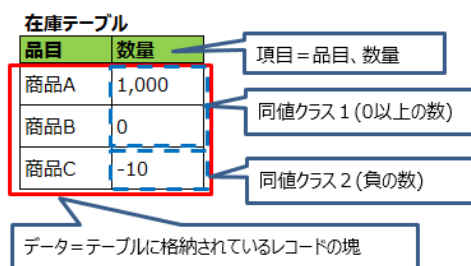


図1 同値クラスのデータ

システム変更によって、これまで負の数を許容していなかった項目に、負の数を許容するなど、新たな同値クラスのデータが発生する修正を行うとする。この場合、そのデータを参照している既存機能に対して、新たな同値クラスのデータの発生にともない、修正が必要か否かを確認する必要がある。対象のシステムの仕様と設計内容の知識が浅い設計者が影響分析を実施すると、本来修正すべき機能について影響なしと判断してしまい、修正の対象から漏れてしまう。この問題を解決するために、システムの仕様と設計内容の知識が浅い設計者でも利用できるような影響分析手法を検討した。

手法の検討にあたり、修正すべき機能の抽出が漏れてしまう理由について考察した。その結果、非正規化によって同じ項目が複数のテーブルに定義されている場合、複数のテー

第5分科会 (Aグループ)

ブルの項目を調査対象としなければならない、影響分析の作業が複雑になるためではないかと仮定した。本研究では、この仮定をもとに影響分析手法を考案した。

以降では、第2章において本研究の背景を述べる。第3章で、先行研究や、課題の構造を踏まえて考案した手法について説明する。第4章では手法の検証結果について報告し、最後に第5章で本研究の成果と課題をまとめる。

2. 背景

2.1 課題

システム変更によって、新たな同値クラスのデータが発生した場合、そのデータを参照している既存機能に対して、修正が必要か否かを確認しなければならない。対象のシステムの仕様と設計内容の知識が浅い設計者が影響分析を実施すると、本来修正すべき機能に対して影響なしと判断してしまう。既存機能の抽出漏れは、レビューや回帰テストによって検知する必要があるが、システム変更にとまって作成された設計書だけでは、影響範囲に漏れがあるかどうかの確認はできない。よって、システム全体の仕様と設計内容を把握している担当者に、レビューや回帰テストケースの作成の負荷が集中してしまい、以下のような課題が発生する。

1. 優先度の高いシステム変更が、特定の担当者の負荷を理由に、後回しになる
2. 担当者の突然の離脱によって、十分なレビューや回帰テストが実施できず、システム変更による欠陥が増えてしまう
3. 優秀な担当者に残業が集中することで、不公平感が生じてしまう

2.2 影響分析に関する考察

図2は、受注・発注などの業務を想定したER (Entity Relationship) 図である。記載方法は、IDEF1X (Integration Definition) 記法に準拠している。実在のシステムのものではなく、本論文用に簡略化している。

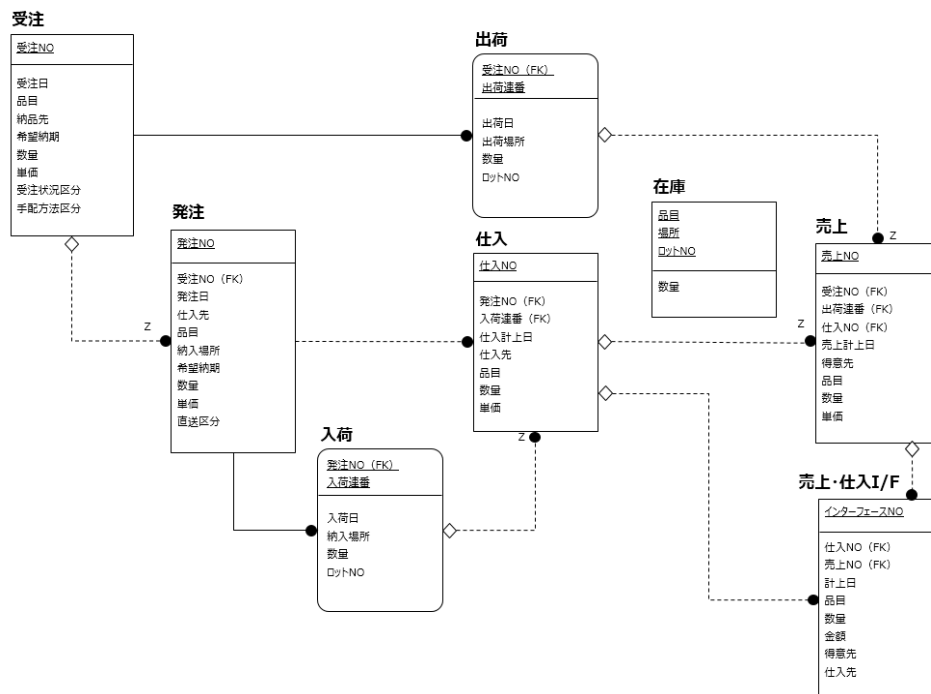


図2 受注・発注などの業務を想定したER図

図2では、業務要件を実現するための非正規化が行われ、複数のテーブルに同じ項目が定義されている。表1に具体例を示す。

表 1 非正規化を行うことがある設計パターン

No	非正規化を行うことがある設計パターン	ER図における具体例
1	外部キーによって定義される参照先のテーブルのレコードが、0件であることを許容する	受注テーブルと発注テーブルの品目・数量
2	先行業務のテーブルのデータを分割する	受注テーブルと出荷テーブルの数量
3	参照先のテーブルの値をデフォルト値として引き継ぐが、後続の業務で変更可能とする	発注テーブルと仕入テーブルの単価
4	他システムへ送信するためのデータ (I/Fデータ) を、独立したテーブルで管理する	仕入テーブルと売上・仕入I/Fテーブルの各項目
5	参照時のレスポンス向上のために、他のテーブルの数値項目を集計した結果を保持するテーブルを定義する	在庫テーブル (入荷テーブルの数量と出荷テーブルの数量を集計する)

表 1 の No1 のパターンの具体例について説明する。受注した品目を仕入先に発注し、自社に入荷せず納品先へ直送するような業務（以降、直送業務と呼ぶ）と、受注には紐づかない発注を行う業務がある。それぞれの業務のデータフロー図を図 3 に示す。

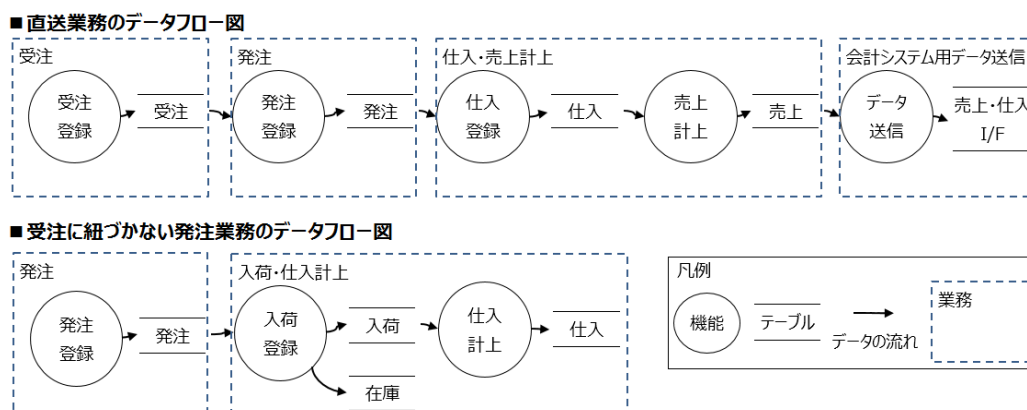


図 3 直送業務および、受注に紐づかない発注業務のデータフロー図

直送業務の場合、受注データをもとに発注データが作成されるため、受注テーブルと発注テーブルは受注 No によるリレーションが定義されている。ただし、受注を伴わない発注業務も存在するため、発注テーブルに対応する受注テーブルの件数が 0 件となる場合もある。よって、受注テーブルと発注テーブルの両方に品目・数量を保持し、直送業務の場合には、両方のテーブルに同じ値を設定するような設計としている。

非正規化が行われている場合、あるテーブルの項目に新たな同値クラスのデータが発生すると、そのデータが受け渡される他のテーブルにおいても、新たな同値クラスのデータが発生する可能性がある。よって、影響分析において複数のテーブルを調査対象とする必要があるため、影響分析の作業が複雑になるといった弊害が生じる。

2.3 先行研究

2.3.1 他システムに対する影響分析

システム変更に伴う他システムへの影響を調査する手法に関する先行研究[1]では、システム変更時に、関連する他システムの概要を記載したドキュメントを参照する仕組みを構築することで、設計者に他システムへの影響の気づきを促している。ただし、新たな同値クラスのデータの発生にともなう影響分析に特化しているわけではないため、本研究の課題には適用できない。

先行研究[2]では、DB に登録されるデータ項目間の関係性の変化や、データバランスの変化のようなデータ傾向が変化した場合における、影響分析を研究の対象としている。設計者が、その変化に気づけるようにするために、「データベース変更チェックポイント (DB-CCP : Database Change Check Point)」を定義している。修正時に作成するトレーサビリティマトリクス（以降、TM と呼ぶ）[3]に DB-CCP を組み込むことで、設計者に、他システムに及ぼす影響への気づきを促すことを狙っている。影響が波及する他システムの調

第5分科会 (Aグループ)

査方法は、ER図やI/F仕様書（他システムへデータを送信する機能の仕様書）といったドキュメントをもとに抽出することを想定している。しかし、TMの縦軸を、他システム及びI/Fデータが格納されるテーブル（以降、I/Fテーブルと呼ぶ）としているため、I/Fテーブルを定義する以外の理由で非正規化が行われている場合、先行研究[2]は適用できない。

2.3.2 テーブル間のリレーションを利用した影響分析

ER図で表現されるリレーションをもとにテーブル間の関係を特定し、複数のテーブルを対象として、影響分析を行う先行研究[4]がある。[4]において開発された影響分析用システムでは、直接リレーションを定義した項目の場合は、複数のテーブルを対象として影響分析が可能である。ただし、直接リレーションが定義されていない項目の場合は、複数のテーブルを対象とした影響分析を実施することはできず、非正規化によって複数のテーブルに定義された項目への影響を見逃してしまう。

具体例としては、図2における受注テーブルの受注NOの桁数変更のような修正であれば、リレーションが定義されている発注テーブルや売上テーブルの受注NOへの影響を、特定できる。ただし、他テーブルとのリレーションが定義されていない受注テーブルの数量に、システム変更前は許容していなかった0以下の数を許容するといった場合に、発注テーブルの数量にも影響があることを検知することはできない。

2.3.3 先行研究と比較した本研究の位置づけ

新たな同値クラスのデータが発生していたにも関わらず、それを検知できない場合、他のテーブルに対する調査自体が行われず、影響する機能の抽出を漏らしてしまう恐れがある。新たな同値クラスのデータが発生をともなうか否かを、担当者に依存せず確認できるようにするための手法として、DB-CCPは有効である。よって、本研究で提案する手法においても、修正がDB-CCPに該当するか否かのチェックを行う、という手順を取り入れた。

影響が波及する機能の抽出方法については、非正規化された項目に新たな同値クラスのデータが発生した場合に対応するため、複数のテーブル間における項目の対応関係を表現できるような影響分析手法を、新たに検討した。

3. 提案手法について

3.1 DB-CCPの再定義

先行研究によって定義されたDB-CCPは、先行研究で分析したシステムの過去事例をベースとしており、あくまで固有システムの一例である。本研究ではシステム変更によるプログラム修正の影響分析を対象としているが、先行研究ではプログラム修正を伴わない影響分析も対象としている（店舗マスタの登録内容の変更によって、他システムのレスポンスに悪影響を及ぼさないかなど）。そのため、本研究では、研究員の経験をもとに、プログラムの分岐での使用頻度が高いであろうパターンを洗い出し、DB-CCPを再定義した（表2）。

表2 本研究で再定義したDB-CCP

No	DB-CCP	影響度分析方法
1	主キー制約、一意キー制約の変更	該当のテーブルを参照している機能を調査
2	テーブル間のリレーションの多重度の変更	該当のテーブルを参照している機能を調査
3	処理の分岐で使用されるコード値の組合せパターンを増やす変更	該当の項目を参照している機能を調査
4	数値項目に対して、これまで発生し得なかった符号の値を許容する変更 あるいは0が許容されなかった項目に0を許容する変更	該当の項目を参照している機能を調査
5	数値・日付項目に対して、他の項目との大小関係が崩れるようなデータを許容する変更	該当の項目を参照している機能を調査
6	NOT NULL項目にNULLを許容する変更	該当の項目を参照している機能を調査

表2のDB-CCPが有効かどうかを評価するために、研究員の所属するプロジェクトにおける10件の過去事例に適用した。適用した過去事例は、システム変更前の機能が想定して

第5分科会 (Aグループ)

いないデータが発生した際に、修正が必要な機能の抽出が漏れていたシステム変更である。その結果、全ての事例が表 2 の DB-CCP に当てはまったため、表 2 の DB-CCP は有効であると判断した。よって、本研究の DB-CCP としては、表 2 を提案する。

表 2 では、それぞれの DB-CCP に対して、どのような影響分析を実施すべきかについても記載している。DB-CCP のうち、表 2 の No1, 2 は、該当のテーブルのデータを抽出している機能において、想定している件数と異なる件数のデータが抽出される可能性がある変更となる。そのため、影響分析では、テーブル名をもとに該当のテーブルを参照している機能を特定し、データの抽出方法を確認する。

一方で表 2 の No3, 4, 5, 6 は、特定の項目において新たな同値クラスのデータが発生する変更となる。そのため、影響分析では、テーブル名をもとに該当のテーブルを参照している機能を特定した後に、該当項目の利用有無を確認する。また、その項目の値が他のテーブルの項目に受け渡されていないかといった調査も実施し、受け渡されている場合は、受け渡し先のテーブルに対して同様の調査を実施する。

3.2 影響が波及する機能の抽出方法

3.2.1 CRUD 図

DB-CCP によって新たな同値クラスのデータが発生したことは検知できるが、その発生がどの機能に影響を及ぼすかは、別途調査が必要となる。調査においては、テーブルと機能の関連を表現する CRUD 図が有効である。図 4 は、[5]による表記を参考に、横軸を機能、縦軸をテーブルとして、交点に C (Create)・R (Read)・U (Update) ,D (Delete) を記載した CRUD 図である。

業務		機能		テーブル														
				受注	発注	入荷	仕入	在庫	出荷	売上	売上仕入	I/F						
受注	受注登録	C	R															
	受注取込バッチ	U	D															
発注	発注登録																	
		C	R															
入荷	入荷登録																	
		U	R	C	R													
仕入	仕入登録																	
出荷	出荷登録																	
		U	R															
売上	売上修正																	
会計システム	売上・仕入送信																	
	用データ送信																	
	バッチ																	

図 4 CRUD 図

先行研究[4]においても、影響が波及する機能の抽出に CRUD 図を活用している。また、CRUD 図の縦軸をテーブルではなくプログラムの変数とし、プログラム修正時の影響分析に利用する研究[6]も存在する。表 2 の No1, 2 に関しては、CRUD 図による影響分析を適用可能なため、本研究で新たな影響分析手法の検討は行わない。

表 2 の No3, 4, 5, 6 に関しては、項目の値が他のテーブルに受け渡しされていないかといった調査も必要となるが、図 4 の CRUD 図には項目の情報が表現されていない。ただし、CRUD 図に項目の情報まで表現することを目的とした IRUN 分析[7]と呼ばれる手法がある。IRUN は、Import (挿入), Refer (参照), Update (更新), NULL (空値セット) の頭文字をとったものである。IRUN 分析を行うと、テーブルの項目がどの機能によって参照されるかを表現することができる。ただし、IRUN 分析では、単一のテーブルに定義された項目に着目しているため、他のテーブル項目に影響を及ぼす可能性があるか否かは、確認できない。IRUN 分析を行った CRUD 図を、[7]より引用する (図 5)。

第5分科会 (Aグループ)

プロセス(アクティビティ) 対応する画面		購入商品確定 (ショッピングカート内容確認)	配送・支払確定 (レジカウンター)	注文確定 (注文内容確認)
エンティティ名	属性名			
注文内容	注文番号	I		(D)
	商品代金支払方法		I,U	(D)
	商品届け方法		I,U	(D)
	商品購入合計	I,U		(D)
	商品配送料		I,U	(D)
	購入消費税	I,U		(D)
	商品注文年月日	I,U		(D)
	注文総称	I,U		(D)
	商品届け年月日		I,U	(D)
	顧客番号	I		(D)
注文明細	注文番号	I		(D)
	商品コード	I,U		(D)
	商品購入数量	I,U		(D)
	商品購入単価	I,U		(D)

図 5 IRUN 分析を行った CRUD 図

3.2.1 複数のテーブル間の項目対応表

本研究では、複数のテーブル間の項目の対応関係を表現するために「複数のテーブル間の項目対応表」を作成した。表の構造は以下のとおりである。

1. 業務の発生順にテーブルを配置する。1つの業務で複数のテーブルのデータが発生する場合、発生順にテーブルを配置する（例：出荷業務において、出荷データをもとに売上データを同時作成する場合、出荷テーブルの次に売上テーブルを配置する）。
2. 配置したテーブルに対して、項目の値の設定仕様を記載する。先行業務で発生したテーブルのデータをそのまま引き継ぐ場合は、同じ行に項目を配置する。先行業務で発生したテーブルのデータを加工して設定する場合は、その関係に気づけるようにするために、もとの項目の次の行に配置する。もとの項目が複数存在する場合は、最初の項目の次に配置する。一つの項目の値を加工して、複数の項目の値が求まる場合は、テーブル定義の順番に配置する。
3. 他のテーブルの項目と対応関係がないセルは、確認の必要がないため、灰色にする。

図 6 は、受注・出荷・他システム（会計システムを想定）へのデータ送信という業務を対象として作成した、「複数のテーブル間の項目対応表」である。

No	受注		出荷				会計システム用データ送信	
	項目名	設定仕様	項目名	設定仕様	項目名	設定仕様	項目名	設定仕様
1	受注NO	シーケンスで発番	受注NO	受注の値を引き継ぐ	受注NO	出荷の値を引き継ぐ		
2	受注日	画面、I/Fの値						
3	品目	画面、I/Fの値			品目	受注の値を引き継ぐ	品目	売上の値を引き継ぐ
4	納品先	画面、I/Fの値			得意先	納品先マスタを元に設定	得意先	売上の値を引き継ぐ
5								
6	希望納期	画面、I/Fの値						
7			出荷日	画面の値 ※希望納期に間に合う日付を設定する	売上計上日	出荷の値（出荷日）を引き継ぐ	計上日	売上の値を引き継ぐ
8	数量	画面、I/Fの値	数量	受注の値を画面に初期表示する。 分割出荷の場合は、一部数量のみが設定される	数量	出荷の値を引き継ぐ	数量	売上の値を引き継ぐ
9	単価	品目、納品先、希望納期を元に販売単価マスタから取得する			単価	受注の値を引き継ぐ		
10							金額	数量×単価 ※単価は売上テーブルの値
11	受注状況区分	「1：受注」固定						
12	手配方法区分	「1：自社から出荷」固定						
13			出荷運番	分割出荷のたびにカウントアップ	出荷運番	出荷の値を引き継ぐ		
14			出荷場所	画面の値（在庫テーブルの数量を参照しながら設定）				
15			ロットNO	画面の値（在庫テーブルの数量を参照しながら設定）				
16					売上NO	シーケンスで発番	売上NO	売上の値を引き継ぐ
17							インターフェースNO	シーケンスで発番
18							仕入NO	NULL
19							仕入先	NULL

図 6 「複数のテーブル間の項目対応表」

受注した品目を自社から出荷する業務と、図 3 の直送業務のように、異なるパターンで

第5分科会 (Aグループ)

データが流れる場合は、一つの表で表現することができない。そのような場合は、表を分けて作成する(付録Aは直送業務の場合の「複数のテーブル間の項目対応表」である)。図6では、受注テーブルの次に出荷テーブルが記載されており、出荷テーブルの次に売上テーブルが配置されている。付録Aでは受注テーブルの次に発注テーブルが記載されており、発注テーブルの次に仕入テーブルが記載されている。

「複数のテーブル間の項目対応表」の作成で必要となる、業務の発生順および、業務とテーブルの関連は、業務フロー図とデータフロー図をもとに確認する。テーブルと機能の関連はCRUD図で確認し、テーブル項目の設定仕様は各機能の設計書で確認する。

「複数のテーブル間の項目対応表」では、テーブル項目の対応関係を機能の設計書を元に表しているため、リレーションが定義されていない項目においても、対応関係を確認することができる。また、一階層ではなく多階層に渡って表現しているため、非正規化によって複数テーブルに定義されている項目の対応を、横串で確認することができる。そのため、先行研究[4]の課題であった、リレーションが定義されていない項目に対しても、複数のテーブルにわたる影響調査を行うことができる。

3.2.1 影響分析の作業手順

本研究では、影響分析を以下の手順で実施することを提案する(付録Bにフロー図を記載)。「DB-CCP」、「CRUD図」、「複数のテーブル間の項目対応表」は既に存在する前提である。

1. システム変更にとまなう設計書とDB-CCPをもとに、新たな同値クラスのデータが発生する修正か否かを確認する
2. 新たな同値クラスのデータが発生する修正の場合、「複数のテーブル間の項目対応表」をもとに、影響が波及するテーブルと項目を確認する
3. CRUD図をもとに、2.で確認したテーブルを参照する機能を洗い出す
4. 3.で洗い出した機能の設計書において、2.で確認した項目が使用されている場合、修正の必要性を判定する

なお、回帰テストケースを作成する際に本手法を適用する場合は、4.において修正の必要性がないと判定された機能も、候補とすることが望ましい。その理由は、回帰テストは、未変更部分で欠陥が発生しないことの確認を目的としているためである[8]。

4. 提案手法の評価

4.1 評価結果

今回検証対象としたシステムには、282の機能と263のテーブルが存在する。評価にあたりシステム全体の「複数のテーブル間の項目対応表」を作成するのは困難なため、検証にあたっては一部分(7機能, 8テーブル)のみを対象とした。なお、「複数のテーブル間の項目対応表」の作成者は、システムの仕様と設計内容の知識が豊富な設計者である。

評価対象の修正が、どのDB-CCPにマッチするかの確認は、設計レビューや回帰テストケース作成での利用を想定し、機能の設計者以外が実施した。表2のNo5のDB-CCPにマッチした1件のシステム変更を対象とし、システムの仕様と設計内容の知識が浅い担当者が、機能の抽出漏れを検知できるかを評価した結果、1件の漏れを検知できた。

具体的には、リレーションが定義されている2テーブル間の数量の大小関係が崩れるような、新たな同値クラスのデータが発生したシステム変更を対象とした。そのデータをもとに作成される他のテーブルを参照する機能に対して、設計が漏れていたことを検知できた。ただし、手法の適用事例が少ないため、機能の抽出漏れの検知が本手法によるものであると断定することはできない。本手法を適用しなくても漏れを検知できた可能性がある。

4.2 今後の課題

「複数のテーブル間の項目対応表」では、複数テーブルに定義されている項目の対応関

第5分科会 (Aグループ)

係を一つの表で表現しているため、複数テーブルの項目の対応関係を一目で確認できる。ただし、二桁以上のテーブルが存在する場合、一つの表として対応関係を表現することは難しい。また、テーブル項目の設定仕様を他の設計書から転記しているため、いわば設計情報の非正規化を実施していることになり、仕様変更時の設計書のメンテナンス工数が増加する。よって、二桁以上のテーブルが存在するシステムや、頻繁に仕様変更が行われるシステムでは、本手法の適用は難しいと考える。

この課題の解決には、先行研究[4]が開発している影響分析用システムが参考になる。影響分析の用途であれば、テーブルの全項目の対応関係を一つの表で表現する必要はなく、新たな同値クラスが発生した項目に対してのみ、テーブル間の項目の対応関係が参照できれば良い。付録Cに、先行研究[4]の影響分析システムの画面イメージを引用する。この影響分析用システムでは、テーブルの項目を検索条件として指定し、その項目と他のテーブルの項目の対応関係を確認できる。

5. おわりに

本研究では、複数のテーブルに定義される項目の存在が、影響分析の作業を複雑にする原因であると仮定し、新たな影響分析手法を提案した。提案した手法を1件のシステム変更に応用した結果、1件の機能の抽出漏れを検知することができた。ただし、本研究には「4.2 今後の課題」で述べたような未解決の課題が残る。

今後、テーブル間の項目の対応関係を表現するために適した設計書フォーマットを考案し、新たにシステムを開発するプロジェクトにおいて適用したい。設計書をツールで解析し、必要な情報を抽出することで、影響分析システムの開発につなげることができる。設計書をツールで解析した結果を利用し、設計の品質を高めることを目的とした先行研究[9][10]が存在する。これらの先行研究も参考に、課題の解決を図っていきたい。

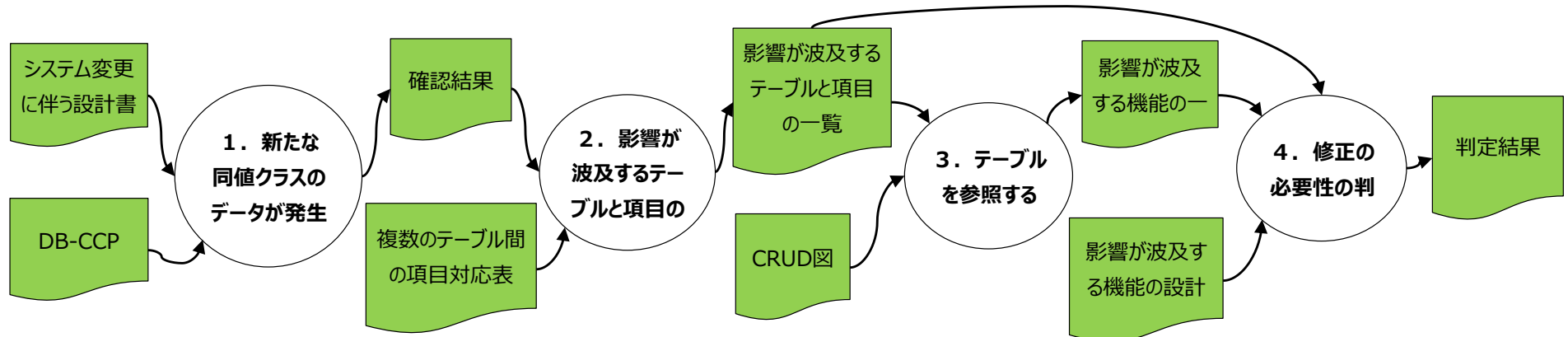
参考文献

- [1] 木下 良介, 中澤 康郎, 大杉 仁司, 変更依頼の対応箇所を検討する前に他システムへの影響を検知する方法, 日本科学技術連盟 ソフトウェア品質管理研究会 2011年度第6分科会成果報告, 2012
- [2] 林 慎一郎, 藤井 伸之, データベースを共有し合うシステム間での変更の影響を効果的に検知する方法, 日本科学技術連盟 ソフトウェア品質管理研究会 2014年度第6分科会成果報告, 2015
- [3] 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, 2007
- [4] 稗方 和夫, 大和 裕幸, 深田 直人, 中村 覚, 岡田 伊策, 齋藤 稔, 笈田 佳彰, 渡辺 郁雄, 松本 滋, システムの仕様変更調査における設計内容を用いた影響分析システムの開発, 日本機械学会第24回設計工学・システム部門講演会講演論文集, 2014
- [5] 独立行政法人 情報処理推進機構 ソフトウェアエンジニアリングセンター, 発注者ビューガイドライン データモデル編, pp. 第1部-23, 2008
- [6] 加藤 正恭, 小川 秀, CRUDマトリクスを用いたソフトウェア設計影響分析手法, 情報処理学会第73回全国大会, vol.2011 no.1, pp.249-251, 2011
- [7] 真野 正, 実践的データモデリング入門, 翔泳社, pp.143, 2003
- [8] 大西 建児, 勝亦 匡秀, 佐々木 方規, 鈴木 三紀夫, 中野 直樹, 町田 欣史, 湯本 剛, 吉澤 智美, ソフトウェアテスト教科書 JSTQB Foundation 第3版, 翔泳社, pp.74, 2011
- [9] 濱野 義満, 銀林 純, 業務フロー図に含まれる仕様を形式表現する手法の提案について, 2014年度 情報処理学会関西支部 支部大会 講演論文集, 2014
- [10] 元山 厚, 中谷 多哉子, 設計項目間の不整合を検出するためのシステム開発と評価, 情報処理学会デジタルプラクティス, Vol.5 No.3, pp.239-248, 2014

付録A 直送業務の場合の「複数のテーブル間の項目対応表」

受注		発注		仕入・売上				会計システム用データ送信				
受注テーブル		発注テーブル		仕入テーブル		売上テーブル		売上・仕入I/Fテーブル（仕入）				
								仕入		売上		
No	項目名	設定仕様	項目名	設定仕様	項目名	設定仕様	項目名	設定仕様	項目名	設定仕様	項目名	設定仕様
1	受注NO	シーケンスで発番	受注NO	受注の値を引き継ぐ			受注NO	受注の値を引き継ぐ				
2	受注日	画面、I/Fの値										
3	品目	画面、I/Fの値	品目	受注の値を引き継ぐ	品目	発注の値を引き継ぐ	品目	仕入の値を引き継ぐ	品目	仕入の値を引き継ぐ	品目	売上の値を引き継ぐ
4	納品先	画面、I/Fの値	納入場所	受注の納品先の値を引き継ぐ								
5							得意先	納品先マスタを元に設定	得意先	NULL	得意先	売上の値を引き継ぐ
6	希望納期	画面、I/Fの値	希望納期	受注の値を引き継ぐ								
7			発注日	画面の値 ※希望納期に間に合う日付を設定する								
8	数量	画面、I/Fの値	数量	受注の値を引き継ぐ	数量	発注の値を引き継ぐ	数量	仕入の値を引き継ぐ	数量	仕入の値を引き継ぐ	数量	売上の値を引き継ぐ
9									金額	数量×単価 ※単価は仕入テーブルの値	金額	数量×単価 ※単価は売上テーブルの値
10	単価	品目、納品先、希望納期を元に販売単価マスタから取得する					単価	受注の値を引き継ぐ 訂正機能によって発注時と異なる単価に変更することができる				
11	受注状況区分	「1：受注」固定										
12	手配方法区分	「2：直送取引」固定										
13			発注NO	シーケンスで発番	発注NO	発注の値を引き継ぐ						
14			仕入先	画面の値	仕入先	発注の値を引き継ぐ			仕入先	仕入の値を引き継ぐ	仕入先	NULL
15			単価	品目、仕入先、希望納期を元に購買単価マスタから取得する	単価	発注の値を引き継ぐ 訂正機能によって発注時と異なる単価に変更することができる						
16			直送区分	「2：直送取引」固定								
17					仕入NO	シーケンスで発番			仕入NO	仕入の値を引き継ぐ	仕入NO	NULL
18					入荷運番	NULL						
19					仕入計上日	画面の値	売上計上日	仕入の値（仕入計上日）を引き継ぐ	計上日	仕入の値（仕入計上日）を引き継ぐ	計上日	売上計上日
20							売上NO	シーケンスで発番	売上NO	NULL	売上NO	売上の値を引き継ぐ
21							出荷運番	NULL				
22									インターフェースNO	シーケンスで発番	インターフェースNO	シーケンスで発番

付録B 影響分析の作業手順



付録C 先行研究[4]の影響分析システムの画面イメージ

検索条件を追加する

項目CRUD検索 (仮称)

項目 :

マスタ・テーブル :

機能 :

all 推論あり 完全一致
 create 推論なし 部分一致
 read 同義語検索
 update
 delete

キーワード

 AND検索 OR検索

シート名

ファイル名