

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

New methodology to identify the impacted area of the existing surrounding systems
in advance before considering detail specification of the change request

主査	清水 吉男	株式会社システムクリエイツ
副主査	飯泉 紀子	株式会社日立ハイテクノロジーズ
アドバイザー	足立 久美	株式会社デンソー
研究員（リーダー）	木下 良介	株式会社リンクレア
	中澤 康郎	株式会社メディカルシステム研究所
	大杉 仁司	東京海上日動システムズ株式会社

研究概要

今日、企業の業務システムは、多数のシステムを用いて多様な業務サービスを提供している。そのような企業のシステム開発の現場では、システム毎に組織や担当者が分かれていることが多い。そのため、担当者の関連システムに対する理解不足から、システム間の連携漏れを発生させている。そこで、企業内の業務サービスに精通し保有システムを横断的に観ることができる部署、担当者が求められている。加えて、顧客からの仕様レベルの変更依頼（具体的な変更指示）が、関連する他システムへの影響を気付かせない一因となっている。

そこで我々は、USDМ (Universal Specification Describing Manner) が持っている仕様から要求を導出する技法を利用し、さらに、関連するシステムの「システムプロファイル」を利用して、他システムへの影響の有無を洗い出す「変更要求-関連システム表」を考案した。これを利用することで、他システムへの連携漏れを防げるようになる。

Abstract

Our team established a new methodology accompanying with a newly created template named "Change request base related system identifying tool" which utilizes one of the techniques of USDМ (Universal Specification Describing Manner), a methodology how to extract and define the user request from a specification, in order to help system engineers identifying the impacted area of the surrounding systems, which would be used referring the system profile of the existing system. By introducing this solution, system integrators can exhaustively identify the integration needs for the surrounding systems.

Nowadays, commercial operation systems provide extensive services using multiple systems. In such a situation, each system has different assigned system engineers or departments. This introduces a lot of opportunities of the oversight in the area of system integration as the assigned system engineers know little about the related surrounding systems. As a result, the needs for having the expert system engineer / department that can comprehensively views all existing systems with the broad range of the knowledge about internal services and operations have been increasing. In addition, the difficulty of identifying the impact to the related systems is increasing as the level of the granularity and specifications requested by customers are becoming fine and detail, which sometimes leads the developer to overlook the impact to the other system components.

1. 研究動機

1.1. 現状認識

今日、エンタープライズ系業務システムは、多数のシステムによって構成され、企業内、企業間の複雑な業務を実現している。また近年、組織改変や体制の細分化に伴い、組織や業務ごとのシステムが開発され、システム部門は担当部門や担当者を各システムに割り当ててきた。その結果、各々のシステム担当部門は、自分達が担当するシステムには精通しているが、連携している他システムについては知識が乏しい状態となっている。

その様な環境で派生開発を進める際、顧客からの変更依頼に対し、他システムへの影響を調査する事なく進めてしまう状況が発生している。その結果、担当のシステムは問題なく動作するが、全く手を加えていない他システムで不具合が起きてしまう。多くの場合、それらが判明するのは、開発工程後半のシステムテストまたはシステム稼働後であり、担当者達を悩ませることとなる。例えば顧客から「あるシステムで使用している区分を変更する」という変更依頼が来た場合、担当者は他システムの影響等を考慮せずに、自システムの区分のみを対応してしまう。しかし、その区分は他システムでも参照されていることに気付いていないため、本番稼働後に他システムで不具合が発生し、顧客に損害を与えてしまう。

最近では、クラウドやグリッドを利用したシステム連携による様々な構想が雑誌等で取り上げられており、今後、益々複雑なシステム構成になっていく。従って、システムを跨いだ組織・担当者間の連携不足による不具合を防ぐ仕組みが必要である。

1.2. 問題点

我々は現状認識で述べた内容を分析するにあたり、各研究員が関わっているプロジェクトの過去の不具合からこの問題に該当する事例を抽出し、解決の手掛かりを得るために先行技術である XDDP[1]を参照した。

XDDP によると、顧客からのシステムに対する変更依頼には、要求レベルと仕様レベルの2つが混在[2]しているという。要求レベルでの変更依頼は、システムへの「実現してほしい事」が変更内容として定義されるものとなっている。また、仕様レベルでの変更依頼では、変更箇所がイメージできるほど具体的に示されており、例えば1つの機能にある部品について「こう変更してほしい」と定義されたものとなっている。そして、仕様レベルで届いた変更依頼というのは、他の機能への影響に気付きにくいとされている。

我々は、収集した事例を調べることで、変更依頼の内容が要求レベルとして依頼されているのか、仕様レベルとして依頼されているのかを全く意識せずに変更対応を行っている事に気付いた。そのため、本来、依頼内容の背景にある要求を見つけ出すことができず、他システムへ別途対応が必要であったことを見逃すこととなっているのではないかと推測した。事例を分析するにあたり、変更要求と変更仕様が意識されないことによって不具合が発生させる3つの型に分類した。

(1)変更要求欠漏型

顧客からの変更依頼を分析して変更要求を抽出する際に、他のシステムへの変更要求が漏れてしまった。そのため、必要な変更仕様が漏れて不具合となる。(図1)

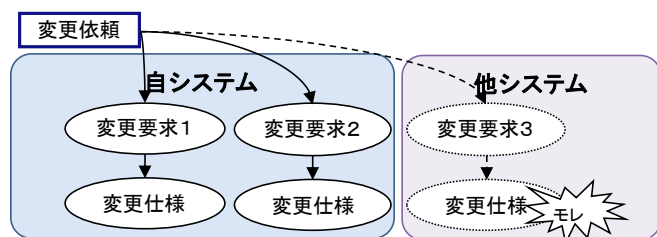


図1: 変更要求欠漏型

(2)変更仕様欠漏型

顧客からの変更依頼が要求レベルで提示され、抽出した変更仕様他システムでも対応が必要であることを気付かない。(図2)

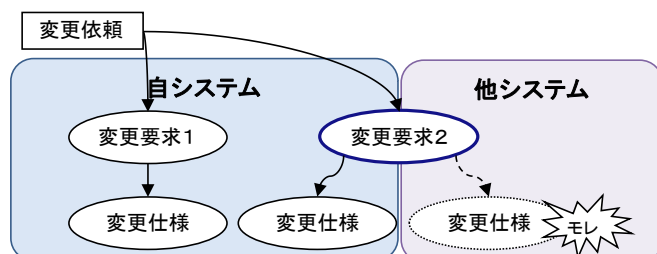


図2: 変更仕様欠漏型

(3)背景無頓着型

顧客からの仕様レベルで提示された変更依頼のみを対応している。その結果、対応依頼の背景にある変更要求に他システムへの変更仕様が含まれていることに気付かない。(図3)

以上の分類を研究員の持ち寄った不具合事例に適用したところ、18 事例の内、6 件が他システムに不具合を出した事例であり、変更要求欠漏型は1件、変更仕様欠漏型は1件、背景無頓着型は4件となった。

特に背景無頓着型については、変更依頼の内容が「仕様レベル」であることを認識することができなければ、その変更がどのような目的によって変更されるのかを読み取ることができず、他システムへの影響有無を検討するといった段階まで踏み込むことができないため問題が根深い。よって、本稿では背景無頓着型について解決する方法を中心に研究対象とした。

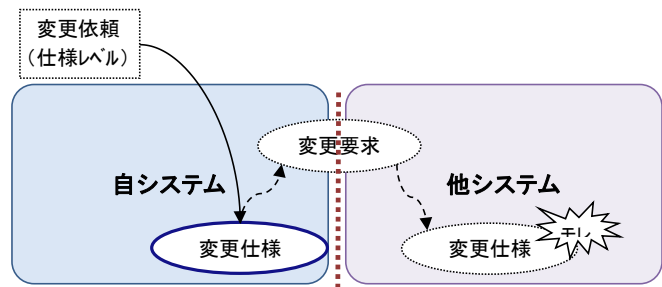


図3: 背景無頓着型

1.3. 原因

仕様レベルの変更依頼を受けた担当者は、なぜ他システムへの影響を認識できないのであろうか。1.1.現状認識で述べたとおり、一定以上の規模の企業の場合、企業活動をサポートするために複数のシステムを連携させながら運営している。また、そのシステム開発にあたっては、複数の組織・担当が分担してプロジェクトを進めている。この認識を踏まえ、研究員の所属する組織で起きた不具合事例を、再度見直してみた結果、以下の課題が明らかになった。

(1)他システムの開発スケジュールが異なっている。

プロジェクト後半に予定されるシステム間の統合テストまでは、各組織の都合により、それぞれのシステムごとの開発スケジュールになっている。そのため、後発でのスケジュールで開発を行っているシステム担当者が、スケジュール上先行しているシステムへの影響する変更内容に気付いた場合、手戻りが発生してしまう。

(2)他システムの業務・機能について情報連携されていない。

関連する組織への自システムの業務・機能等必要な情報を定期的に整理し、連絡するルールになっていない。または、他システムの業務・機能等の情報を収集し組織内で共有する方法を持っていない。そのため、システム間の関係性に着目した資料整備が行われていない。

(3)他システムへの影響確認が修正項目を中心に行っている。

顧客からの変更依頼が、具体的な仕様レベルの変更指示であるため、修正する仕様そのものに関する部分のみの影響確認で終わっている。そのため、他システムに跨って影響しあう仕様についての調査が漏れている。

上記(1)自体は、体制の問題であるが、(2)(3)については、他システムの情報を整理し、変更依頼を受けた段階でその内容を確認することで影響調査依頼ができるような開発プロセスを作成することで問題を軽減することが可能だと考える。また、(1)についても、開発工程の初期段階で調査依頼を行うことができれば、手戻りなどの問題を軽減できる。

2. 先行技術

1.2. 問題点で分類した「背景無頓着型」に起因する問題を解決するために、派生開発で実績のある開発プロセスであるXDDPの技術が使えないかを考えた。

XDDPでは機能変更の開発プロセスにおいて3点セット(変更要求仕様書、トレーザビリティマトリクス(以下、「TM」と記述する)、変更設計書)を使用する。

XDDP で提案されている「TM」の使用方法は、ある1つのシステムへ顧客からの変更要求/仕様が合った場合に、そのシステムの内部の機能グループ間での影響箇所を導き出すことを想定した手法である。我々が課題としている1つの変更仕様からシステムを跨いだ影響箇所を探し出す場合では、どのような機能があるかわからない状況となるため TM を構成できない。

そこで我々はその3点セットの一部である変更要求仕様書の仕組みに着目した。変更要求仕様書は「USDM」[3]の表記方法を踏襲しており、仕様レベルで表現された顧客からの変更依頼に対して、なぜ変更する必要があるのか、その変更にはどのような状況(背景)があるのか、背景を踏まえたうえで何を変更する必要があるのか、という一連のステップを繰り返し「変更要求」を捉える事を目的としている。そこで、この変更要求仕様書の仕組みに、他システムの概要や特徴についての項目を付け加えることで、システムを跨いだ変更仕様による影響を判別できると考えた。

3. 解決策

3.1. 仮説

2. 先行技術で記述した通り、顧客からの変更依頼が仕様レベルで依頼された場合、XDDP では依頼された変更仕様に対して変更要求と理由を検討することにより、影響範囲の視野を広げて判断でき、意識していなかった影響箇所に気付くとされている。そこで我々は、XDDP の手法を参考に次の作業を行うことにより、システムを跨いだ影響についても判断できるのではないかと考えた。

- 仕様レベルで受け取った変更指示から、明示されていない変更要求を導き出す。
(以降、この作業を仕様から要求への「昇華」と記述する。)

- 昇華させた変更要求をもとに、影響が及ぶシステムを検討する(図4)。

始めに仕様を要求に昇華させるための方法を検討した。

XDDP の変更要求仕様書は、変更仕様、変更要求、理由を論理的に整理することができる優れた技術であることが既に知られている。しかし、変更要求仕様書を使用したとしても、今回のケースでは、インプットが変更を施すシステムに対する仕様レベルの文章のみであるため、システムを跨ぐことを認識して、昇華させることは困難であると考えた。そこで、変更要求仕様書を使って仕様を要求へ昇華させる際に、他システムへの変更要求を引き出すヒントとなる「システムプロファイル」を新たに考案した。

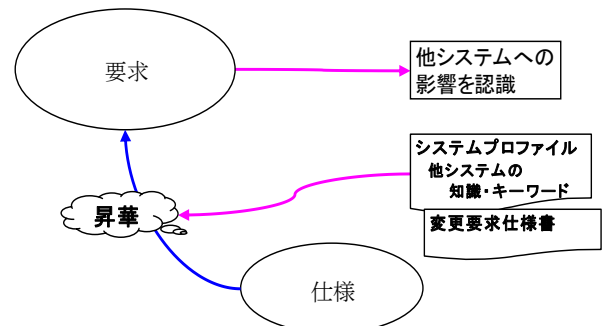


図4 昇華のイメージ

「システムプロファイル」には、今回変更するシス

テムの周辺システム名とその概要を記述し、仕様と共に参照することにより、変更要求を引き出すための気付きを得ようとするものである。また、一般的にシステムは企業内の業務単位に開発されているため、プロフィールを業務単位でシステムと関連づけて記述することにより、昇華させた変更要求が影響するシステムの特定にも役立つのではないかと考えた。

さらに、昇華作業の際、顧客から受け取った仕様と、影響するシステムが表現できるように、変更要求仕様書のフォーマットへ次の2点を追加し、「変更要求-関連システム表」と名付けた。

- 顧客から受け取った変更依頼の原文を記述する欄
- 「システムプロファイル」に記述した周辺システム名と、その概要を記述する欄

※) 本稿のテーマでは昇華した変更要求から変更仕様を引き出す作業は扱わない。そのため、本稿では変更要求仕様書の仕様を記述する欄は省略する。

実際のフォーマットは「図7 変更要求-関連システム表のモデル」に記述する。

【システムプロフィール】

「システムプロフィール」のモデルを以下に記述する(図5)。

システムは一般的に、業務に対して割り当てられていると考え、周辺システム名、システムが担う業務、周辺システムの概要を並べて記述する。実際の作成例は付録B 付表1を参照。

システム	業務	概要
Aシステム	A業務	Aシステムの概要(要求を引き出すため/影響を特定するためのキーワード)
Bシステム	B業務	Bシステムの概要(要求を引き出すため/影響を特定するためのキーワード)
.....

図5 システムプロフィールのモデル

【変更要求-関連システム表】

顧客から届いた変更仕様から、「システムプロフィール」を使って変更要求に昇華させる過程で、「変更要求-関連システム表」を作成する。「変更要求-関連システム表」の作成プロセスを図6に、「変更要求-関連システム表」のモデルを図7に示す。

- 顧客から受け取った仕様レベルの変更依頼を依頼内容欄に記述する(図6①)。
- 「システムプロフィール」を参照しながら、仕様を要求に昇華させる(図6①)。
- 昇華させた要求をもとに、各要求の背景を包含する上位要求を記述する(図6①)。
- 上位要求に対し、影響するシステムまたは影響する可能性があるシステムを「システムプロフィール」から抽出し、関連システム欄に記述する(図6①)。
- 関連システム欄の影響チェック欄に、影響あり/なしを「○」または「×」で記述する(図6①)。

※)b, cの昇華作業の際、不明点は、顧客またはシステム側の受け付け担当者に確認する(図6②・③)。

eで関連システムが影響する可能性はあるものの、実際に影響するか不明の場合は、調査対象として担当部署に調査を依頼し、回答を得る(図6④・⑤・⑥)。

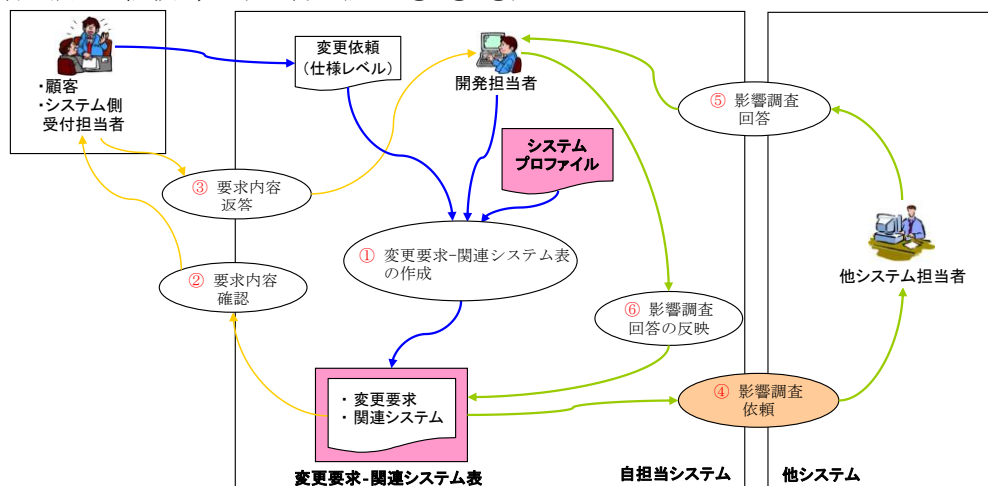


図6 変更要求-関連システム表の作成プロセス

依頼内容						a.
上位要求	1					c.
	理由					
	説明					
関連システム	システム	業務	影響チェック	概要		d.
	Aシステム	A業務	○	Aシステムの概要		
	Bシステム	B業務	要調査	Bシステムの概要		
	＊ ＊ ＊ ＊ ＊	＊ ＊ ＊ ＊ ＊	×	＊ ＊ ＊ ＊ ＊		
変更要求	1.1					b.
	理由	e.				
	説明					

図7 変更要求-関連システム表のモデル

3.2 仮説の検証

考案した「システムプロファイル」と「変更要求-関連システム表」を使用することにより、仕様から要求への昇華を効率的に行うことができ、さらに、昇華させた要求から影響するシステムを特定できないかと仮説を立てた。本節では、この仮説の検証結果について述べる。

(1) シミュレーションの実施

考案した「システムプロファイル」と「変更要求-関連システム表」が有効であるか判断する為、次の2点について検証した。

A. 「システムプロファイル」を使用して変更仕様から変更要求へ昇華できるか

B. 昇華された変更要求が影響するシステムを、「システムプロファイル」の参照により判定できるか

検証はメンバから収集した不具合事例の一部を用いてシミュレーションし、結果を判定した。

また、他システムへの影響判定について、各システム担当者は自分の担当するシステムには精通していることを前提とした。それにより、昇華させた変更要求に対し、影響する可能性があるシステムについては「要調査」と判定された時点で、他システムに調査が依頼され、影響あり／なしの回答が得られるものとした。

使用した「システムプロファイル」を図8に示す(詳細は付録B 付表1を参照)。

業務			システム	概要
大分類	中分類	小分類		
A工場	仕入れ	伝送	伝送仕入システム	各エリアで購入した、部品情報を受信し、DBに登録する。登録された部品情報は、次工程の業務となる仕分けシステム、計測システムに送信される。
		部品データ修正	部品データ修正システム	A工場担当者が、部品情報の変更要求を受けた時点で、ホストに登録されている部品情報の変更を行う。
			
	報告	報告書印刷	報告システム	当日仕入れた全部品ロットの計測完了を受けて、ホストに登録された計測結果を部品管理部門向けの報告情報として出力する。
		伝送報告		
B工場			

図8 使用したシステムプロファイル

(2) シミュレーション結果

シミュレーションを実施し、作成した「変更要求-関連システム表」の概要を図9に示す(詳細は付録B 付表2を参照)。

依頼内容	<ul style="list-style-type: none">・ホストシステムは部品を仕入れた工場（A工場・B工場）を、「管轄コード」にて識別するが、地方工場は、新たに「リンク情報」も付加して識別する。・部品の仕入れ処理において、地方工場の場合は、「リンク情報」に「地方工場コード」を採用する。・＜地方工場の定義＞ 管轄コード：「0」、リンク情報：[地方工場コードを意味する数値] （管轄コード：0 → A工場、1 → B工場）						
上位要求	1	新設する地方工場の部品検査システムは、B工場のシステムをベースに構築。今回稼動する地方工場の仕入れは便宜的にホストの既存環境（A工場環境）を利用する。					
	理由	従来は、工場別にホストシステムのオンライン環境を作成していた。新たなオンライン環境作成はホストシステムの資源圧迫にもつながる為、今回はA工場の環境を利用する。					
	説明						
関連システム	業務			システム	影響チェック	概要	
	大分類	中分類	小分類				
	A工場	仕入れ	伝送	伝送仕入システム	要調査	各エリアで購入した、部品情報を受信し、DBに登録する。登録された部品情報は、次工程の業務となる仕分けシステム、計測システムに送信される。	
			・・・	・・・	要調査	・・・	
			・・・	・・・	要調査	・・・	
		報告	報告書印刷	報告システム	要調査	当日仕入れた全部品ロットの計測完了を受けて、ホストに登録された計測結果を部品管理部門向けの報告情報として出力する。	
			伝送報告		要調査		
変更要求	1.1	・・・					
変更要求	1.2	新たなカテゴリである地方工場について、検査情報を、既存のホストA工場環境に投入するが、今後、他の地方工場が新設された場合、どの地方工場で仕入れられた部品であるかを識別したい（「管轄コード」と「リンク情報」で識別）。					
	理由	<ul style="list-style-type: none">・A工場のシステムを運用する際、各地方工場ごとに処理内容を分けるため。 （システムで識別の必要がある処理は??? → 要調査）・管轄コード、リンク情報のA工場部品検査システムでの使用用途は??? → 要調査・A工場部品検査システムでは管轄コード、リンク情報を何に使用しているか??? → 要調査					
	説明						
変更要求	1.3	・・・					

図9 シミュレーション結果 変更要求-関連システム表

A. 「システムプロファイル」を使用して変更仕様から変更要求へ昇華できるか

シミュレーションを行った結果、「システムプロファイル」のみを参照しただけでは、仕様を要求に昇華させることができなかった。代わりに、昇華作業を行う中で、変更仕様が記述されている依頼内容から、疑問点を単語レベルで抽出し、回答を検討する作業を行った(付録B 付表3)。回答が埋まらない箇所については、有識者に確認して記述した。その結果として作成された疑問点の一覧を見ることにより、変更要求に昇華することができ、さらに、「システムプロファイル」を参照することにより、変更の理由をイメージしやすくなった。

また、理由欄には、疑問点の一覧から引き継がれた未解決の疑問が含まれている。その記述があることによって、変更要求が他システムに対する調査対象であることを認識しやすくなった。

B. 昇華された変更要求が影響するシステムを、「システムプロファイル」の参照により判定できるか

昇華された変更要求と「システムプロファイル」を参照しながら影響システムを判断したところ、業務の名称をキーワードにして影響する可能性がある複数のシステムを調査対象として判別することができた。しかし、複数あるシステムの中から明確に1つのシステムを特定することはできなかった。

また、関連システム欄の記述方式については、当初想定していた要求の上部に記述するよりも、XDDPで使用する「TM」のように横方向に記述した方が、各変更要求に対して影響するシステムを表現し易いことが分かった(付録B 付表4)。

4. 考察

他システムに詳しくない担当者が、「システムプロファイル」を参照することで、影響する他システムを抽出できることは分かった。しかし、影響の有無を確実に判断するためには、業務概要レベルの記載内容だけでは難しく、より詳細なレベル、もしくは判断するのに必要な項目群が更に必要と考えられる。「システムプロファイル」の業務概要の記入にあたって、業務／機能だけでなく、利用者(オペレーションの実施者、当該システムによるサービスの受益者)や、利用データ、主要アウトプット、作業サイクルを盛り込むことで、より判別が容易になると考えている。

今回、「変更要求-関連システム表」という形式を考案したことで、変更要求、変更仕様が明記され、影響する他システムが記録として残る。これを繰り返すことで、影響を判断するための知識が蓄積され、各々の組織に合った形で「システムプロファイル」が成長していくと考える。具体的なプロセスとして、他システムへの調査依頼の結果から新たな気付き(図10 ①)を得た際、または、本稿記載の取り組みを実施したにも関わらず、問題が発生した場合には原因を分析し、今後、同様の問題を発生させないための表現を見いだし(図10 ②)「システムプロファイル」の見直しを行う(図10 ③)。また、「システムプロファイル」の形骸化を防ぐために、定期的に他システムの変更状況を確認し、変更があった時は「システムプロファイル」に変更内容を反映し、最新版の利用を徹底する必要がある(図10 ④)。こうして「システムプロファイル」をメンテナンスすることにより、他システムへの理解が深まり、複数のシステムを跨いだ不具合が減少していくと考える。

本稿では、「背景無頓着型」を中心に解決方法を研究したが、その他2つの型においても他システムに関する知識が不足している事情は同じである。「システムプロファイル」を参考にすることで、今まで気がつかなかった「変更要求」と他システムとの関連性に気づくことで、他システムへの影響を調査しなかったために発生する不具合を減少できると考える。

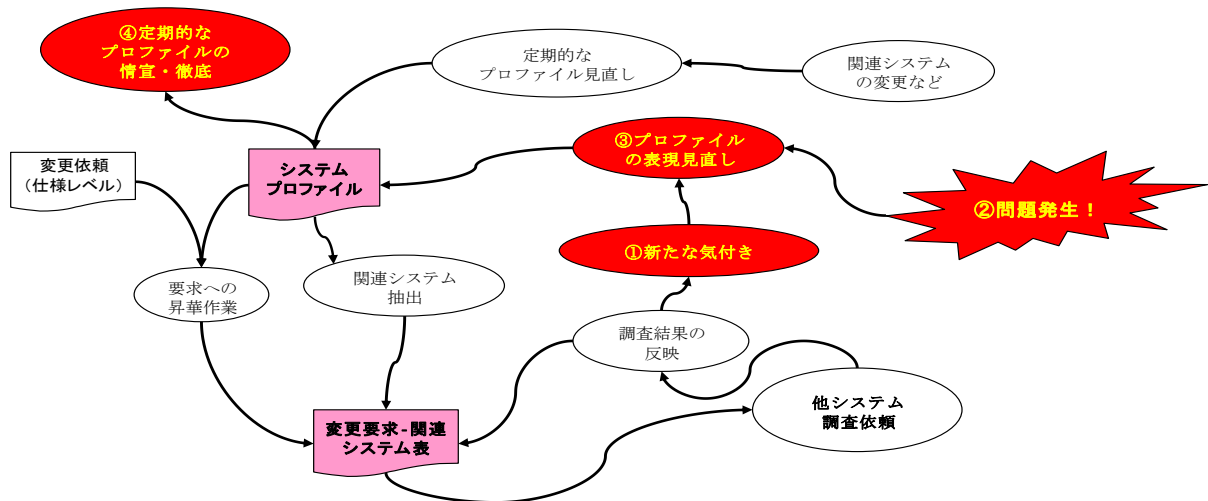


図 10 システムプロフィールの維持・改善プロセス

5. 結論

5.1 結果

仕様レベルの変更依頼が、他システムへの影響を調査せずに進めてしまう事により発生する問題について、本稿では「変更要求-関連システム表」、「システムプロフィール」を用いる事で他システムへの影響に気付き、必要となる調査依頼を出すことができる手法を考案した。

変更依頼内容から変更要求へ昇華させる方法として、「システムプロフィール」のみを参照しただけでは、うまくいかないことが解った。しかし、依頼内容から疑問点を単語レベルで抽出し、その内容を検討する際に、「システムプロフィール」を用いるなど改良を加えることによって、他システムへの影響の気づき、必要となる調査依頼を出せることがわかった。

本稿で提案する「システムプロフィール」を実際に作成するにあたっては、関連する他システムを担当する組織の協力が不可欠である。1.1.現状認識で述べた通り、他システムへの理解が不足した状態で「システムプロフィール」を作成しても、判断に必要な情報が記載されないからである。

5.2 今後の課題

我々は「システムプロフィール」について、実プロセスへ適用させることで、成長していく可能性を秘めたものだと考えている。

他システムの「システムプロフィール」を作成するにあたっての項目と概要の内容については提案に留まっているが、今後、よりシステム担当者が気づきを得やすい表現とはどのようなものかを検討していく必要がある。

また、関連する他システムが数十、数百と大規模な場合について、今回の手法の考え方が通用するかを検証し、改善していく必要があると考えている。

6. 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, 2007
- [2] 清水吉男: SQIP 第6 分科会[派生開発]ミニ講演資料3 版 派生開発プロセス[XDDP]のポイント, 2011
- [3] 清水 吉男:【改定第2版】[入門+実践] 要求を仕様化する技術・表現する技術, 2010

付録 A

付表 1: 事例集(抜粋)

No	結果として困った事実	要求, 仕様 実際の実例(実文)	どの部分でダメだったか (変更要求欠漏型, 変更仕様欠漏型, 背景無頓着型をベースに具体的に記述する. 内容を理解ができるように)	なぜ, そういう表現にするのか (背景・・・制度, 個人差, 書き方)
2	空売り注文のトレーダの計らい指値注文分を成り行き注文へ訂正する際に, 大量の訂正トランザクションが発生し, 高頻度トランザクション検知機能に抵触し停止した. この間, 大量のトランザクションが発生し発注基盤のパフォーマンスに影響を及ぼした.	XXX への空売り注文対応. 成り行き注文への訂正する対応の方針は発注戦略機能システムが発注する注文のみで, トレーダの計らい指値注文に対しての言及は, 価格を訂正すること以外に特に記述はなかった. ユーザとの仕様を決定するために記載したメモは残っているが, そのメモには上記のように, 成り行き注文への訂正でトレーダの計らい指値注文は価格を訂正する旨の記載があるのみ.	変更要求欠漏型 変更依頼に対して要求を作成する際, パフォーマンスによる影響を考慮出来なかった. トレーダの計らい指値注文は, 発注戦略機能の注文のように API 経由で通知されるものではなく, Query (SQL の様な問合せを必要とする電文) で受信するため, 訂正の正常応答電文が遅い. その部分を考慮する記載が一切なかったため, 単純に開発を行ってしまったと思う.	・空売りという注文の特異性にばかり気を取られ, 正しく機能するかどうか重点を置いてしまっている. ・複数のトレーダの計らい指値注文が成り行き注文への訂正時に残っていることや, 発注していること自体がレアケースであること.
3	画面上に, 作業内容とともに当該作業を実施した担当者名を表示する機能があるが, 一部の担当者名が表示されない状態が発生した. ・表示が正しくされない担当者は, 組織変更に伴いシステムへの「ログインID」体系を変更した組織に属する担当者であった. ・担当者名の表示は, 別項目の「担当者コー	・「ログインID」体系の変更については, 切り替えタイミングと, それに伴う各種顧客のデータバックアップ作業についての記述のみ. ・「担当者コード」に関する記述 各種担当者名表示に利用する担当者コードは, 変更しない.	背景無頓着型 「ログインID」から「担当者コード」を求め, 担当者名を表示する機能に気付かなかった. ・担当者名表示は, 画面上入力した「担当者コード」からし	・当該プロジェクトにおいては, 「ログインID」体系の変更は, システム面において, 顧客側の登録作業のみと考えられており, また, 新設体系ではなく, 従来から存在する体系への移行であった. ・担当者名の表示は, 初期の

	ド」をユーザが画面上入力することで、担当者名を引き込む仕様になっている。		かデコードされないと思い込んでいた。 ・「担当者コード」の体系を変更しなかったため、それ以上、当該項目に関する調査を行わなかった。	設計では、顧客が画面上入力した「担当者コード」からのデコードだけを対象としていた。
4	B工場のシステムを移植し、地方工場のシステムを構築。 特定部品においては地方工場で計測した値をA工場での検品工程で参照する場合があるが、参照することができなかった。	要求を記述した文書なし。 以下の仕様が記述 ・ホストシステムは部品を仕入れた工場(A工場・B工場)を、「管轄コード」にて識別するが、地方工場は、新たに「リンク情報」も付加して識別する。 ・部品の仕入れ処理において、地方工場の場合は、「リンク情報」に[地方工場コード]を採用する。 ・＜地方工場の定義＞ 管轄コード:「0」, リンク情報:[地方工場コードを意味する数値] (管轄コード:0 → A工場 , 1 → B工場)	背景無頓着型 仕様に対し、A工場検品システムへの影響が無いと認識していた。	他システムには影響ないと認識していたため、他システムに対する調査をおこなわず、変更するシステムの仕様のみを記述した。
10	空売り対応を行ったある発注戦略機能が、特定の条件下にリアルタイムでの無限ループを引き起こす。ある一定の執行、取り消しを繰り返しコンプライアンス上、違反となってしまう状況となったため、トレーダが発注戦略システムを止めた。	出来高縛りの発注戦略機能に空売り注文が執行できるよう変更する。	変更仕様欠漏型 空売りであるため、反対気配値がUpTickでない限り約定させられない。しかし出来高縛りであるため定められた約定を付ける必要がある。そのため約定しない価格へ約定したい数量を執行する。この際、発注できる残りの株数の計算方法や新規執行した注文が市場で有効となるまでのターンアラウンドを考慮に入れた	フレームワークレベルでの空売り対応は施されていたが、発注戦略機能での対応を初めて施したため、数量計算の断片的なケースにのみ注力し、時系列での動作を検証していなかった。

			仕組みが想定出来ていなかった.	
12	商品改定時に、特約名称について変更があったが、契約登録システムのみ名称変更の手当てがなされ、周辺パッケージの一部の名称表示が旧名称のまま表示された.	「〇〇特約」を「△△特約」に名称変更する.	背景無頓着型 特約名称変更の通知が、取りまとめサプライヤーから連絡されなかったため、個別にデコード処理を行っているパッケージの対応が漏れた.	特約名称の表記については、取りまとめサプライヤーからの連絡を受け、各パッケージにて対応可否を判断し対応することとなっている.
13	社内オンラインの事故受付登録業務において、ある画面の保存登録を行わないケースで、トランザクションアベンドが発生する.	「△項目」を用いて、基準日を超過した場合は、通知処理を行う.	背景無頓着型 社内登録業務のオペレーションケースから、新設項目の初期設定タイミングを設定したが、別システムから連動されるケースについて考慮が漏れた.	項目の初期設定処理については、システム内の仕様書のみの記載となっており、顧客との合意文書上に表現していない.

付録 B シミュレーション結果

付表 1:シミュレーションで使⽤したシステムプロファイル

業務			システム	概要
大分類	中分類	小分類		
A工場	仕入れ	伝送	伝送仕入システム	各エリアで購⼊した、部品情報を受信し、DBに登録する。登録された部品情報は、次工程の業務となる仕分けシステム、計測システムに送信される。
		伝票読込み	伝票読込みシステム	A工場担当者が搬入された部品伝票をスキャナで読み込み、部品情報をホストに登録する。
		部品データ修正	部品データ修正システム	A工場担当者が、部品情報の変更要求を受けた時点で、ホストに登録されている部品情報の変更を行う。
	仕分け	機械系仕分け	仕分けシステム	JIS規定の部品については、トレイに貼付されているバーコード情報を仕分け装置が読み込み、部品情報の所⾷製品データをもとに対象の生産ライン倉庫に振り分けを行う。
		人間系仕分け		特殊形態の部品については、システムから指示される振り分け情報をもとに人間が、対象の生産ライン倉庫に振り分けを行う。
		サンプル保管		マスタ設定条件でピックアップした部品サンプルを保管する。
	
	計測	機械系計測	計測システム	JIS規定の部品については、サンプリングトレイに貼付されているバーコード情報を計測機器が読み込み、チェック部位の計測を行う。
		人間系計測		特殊形態の部品については、計測機器を使用せず、専⾂職⼈によるチェック部位の計測を行い、結果を計測システムに⼊⼒する。
		合格判定		部品単位で全てのサンプリング対象が計測完了した時点で、対象部品ロットに対する使⽤可否判定を自動で行、判定結果を計測システムに登録する。
	
	報告	報告書印刷	報告システム	当日仕入れた全⼾部品ロットの計測完了を受けて、ホストに登録された計測結果を部品管理部門向けの報告情報として出力する。
		伝送報告		
B工場	仕入れ	伝送	伝送仕入システム	各エリアで購⼊した、部品の情報を受信し、DBに登録する。登録された部品情報は、次工程の業務となる仕分けシステム、計測システムに送信される。
		伝票読込み	伝票読込みシステム	A工場担当者が搬入された部品伝票をスキャナで読み込み、部品情報をホストに登録する。
			
	報告	報告書印刷 伝送報告	報告システム	当日仕入れた全⼾部品ロットの計測完了を受けて、ホストに登録された計測結果を部品管理部門向けの報告情報として出力する。
地方工場	仕入れ	伝送	開発中	
		伝票読込み		
		部品データ修正		
	仕分け	機械系仕分け		
		人間系仕分け		
			
	計測	機械系計測		
		人間系計測		
		合格判定		
			
	報告	報告書印刷		
		伝送報告		

付表 2:シミュレーションで作成した変更要求-関連システム表

依頼内容	・ホストシステムは部品を仕入れた工場（Ａ工場・Ｂ工場）を、「管轄コード」にて識別するが、地方工場は、新たに「リンク情報」も付加して識別する。 ・部品の仕入れ処理において、地方工場の場合は、「リンク情報」に「地方工場コード」を採用する。 ・＜地方工場の定義＞ 管轄コード：「0」、リンク情報：「地方工場コードを意味する数値」 （管轄コード：0 → Ａ工場、1 → Ｂ工場）					
上位要求	1	新設する地方工場の部品検品システムは、Ｂ工場のシステムをベースに構築。 今回稼動する地方工場の仕入れは便宜的にホストの既存環境（Ａ工場環境）を利用する。				
	理由	従来は、工場別にホストシステムのオンライン環境を作成していた。新たなオンライン環境作成はホストシステムの資源圧迫にもつながる為、今回はＡ工場の環境を利用する。				
	説明					
関連システム	業務			システム	影響チェック	概要
	大分類	中分類	小分類			
	Ａ工場	仕入れ	伝送	伝送仕入システム	要調査	各エリアで購入した、部品情報を受信し、ＤＢに登録する。登録された部品情報は、次工程の業務となる仕分けシステム、計測システムに送信される。
			・・・	・・・	要調査	・・・
			・・・	・・・	要調査	・・・
		報告	報告書印刷	報告システム	要調査	当日仕入れた全部品ロットの計測完了を受けて、ホストに登録された計測結果を部品管理部門向けの報告情報として出力する。
			伝送報告		要調査	
変更要求	1.1	現在はＡ工場（本社工場）とＢ工場（〇〇工場）のみしかなく、地方工場が存在しない。初の地方工場設立に合わせ、部品検品システムを稼動させたい。また、今後、同様の地方工場を増やしていきたい。				
	理由	地方工場を設立し、全社レベルで、製品供給能力の向上を図る。				
	説明					
変更要求	1.2	地方工場・部品検品システムのものであるＢ工場システムではホスト環境（Ｂ工場環境）を使用しているが、今回、地方工場の検品情報をホスト投入する際は、既存のＡ工場と同じホスト環境（Ａ工場環境）に投入したい。				
	理由	○地方工場に登録した部品検品情報をホストＡ工場環境でも使用する為。 ○下記処理をホストＡ工場環境で行いたい。 ・仕入れ ・検品 ・報告書印刷 （他にはないか??? → 要確認、要調査）				
	説明	地方工場が増設された後も、ホストの運用とＡ工場の運用は基本的に、現行と同じとしたい。				
変更要求	1.3	新たなカテゴリである地方工場について、検品情報を、既存のホストＡ工場環境に投入するが、今後、他の地方工場が新設された場合、どの地方工場で仕入れられた部品であるかを識別したい（「管轄コード」と「リンク情報」で識別）。				
	理由	・Ａ工場のシステムを運用する際、各地方工場ごとに処理内容を分けるため。 （システムで識別の必要がある処理は??? → 要調査） ・管轄コード、リンク情報のＡ工場部品検品システムでの使用用途は??? → 要調査 ・Ａ工場部品検品システムでは管轄コード、リンク情報を何に使用しているか??? → 要調査				
	説明					

付表 3:昇華作業の中で作成した仕様からの疑問点一覧

① ホストシステムは部品を仕入れた工場（Ａ工場・Ｂ工場）を、「管轄コード」にて識別するが、地方工場は、新たに「リンク情報」も付加して識別する。		
疑問点	回答	要求 疑問点と回答をもとに要求を検討
「Ａ工場」とは何？	本社工場	
「Ｂ工場」とは何？	既存の〇〇工場	
「管轄コード」とは		
Ａ工場ではどの様な使われ方をしているか？	ホストでの環境を識別する（Ａ工場環境） その他の役割は？？？	
Ｂ工場ではどの様な使われ方をしているか？	ホストでの環境を識別する（Ｂ工場環境）	
「管轄コード」はシステム内で何をしているか？	ホスト環境（Ａ工場環境、Ｂ工場環境）に合わせ、設定値を変更することにより、どの環境からの情報かを識別する	
識別して何をしているのか（何の為に識別しているのか）？	仕入れた工場を識別し、処理を分けている？？？ 処理名称は？？？	
「管轄コード」を使用している処理・運用は？	ホスト仕入処理 その他にはないか？？？	
「地方工場」とは何か	新しく地方に新設する工場のこと。 現在はＡ工場（本社工場）とＢ工場（〇〇工場）しか存在しない。	地方工場として初の工場を稼働させたい
「地方工場」を識別してどうするのか？	地方工場を識別する処理がある どの処理か？？？	今後、地方工場を増やしていきたい
新たに「リンク情報」も付加して（リンク情報は現在未使用）について		
何故現在未使用の「リンク情報」を使用するのか？	ホスト環境は既存のＡ工場環境、Ｂ工場環境を使用し、かつ、地方工場を識別する為、「管轄コード」の枝番に相当する新たな区分が必要である為。	地方工場は既存工場で使用しているホスト環境に、部品検品データを登録したい
リンク情報を識別して何をしているのか	管轄コードのみでは地方工場を識別できない為、リンク情報で識別する。	
リンク情報を識別することにより影響のある処理、運用は？	「リンク情報」は現在使用していない為、影響なし。ホスト、Ａ工場の運用・処理がそのまま使えるはず。 確認が必要？？？	
通常は何で識別するのか？	現状はＡ工場、Ｂ工場しかないため、「管轄コード」で識別。今後は「リンク情報」も考慮する。	
なぜ「管轄コード」のみでの識別ではダメなのか？	「管轄コード」はホスト環境（Ａ掘工場境、Ｂ工場環境）に紐づいている。今後、地方工場が増えるたびに当コード値を増やすことはできないから。	

② 部品の仕入れ処理において、地方工場の場合は、「リンク情報」に「地方工場コード」を採用する。		
疑問点	回答	要求
なぜ「仕入れ処理」だけに限定しているのか？	検品情報がどの工場に属するか決定するのが「仕入れ処理」である為。	ホストシステムでどの地方工場で受付けられたかを識別したい
地方工場コードとは何か？	地方工場を示す１桁のコード	

<地方工場の定義> ③ 管轄コード：「0」、リンク情報：[地方工場コードを意味する数値] （管轄コード：0 → Ａ工場、1 → Ｂ工場）		
疑問点	回答	要求
なぜ「管轄コード」を「0」固定にする必要があるか？	ホストでＡ工場環境を使用したいから	今回稼働させる地方基幹工場はホストのＡ工場環境を使用したい
「管轄コード」と「リンク情報」を対にして設定する意味は？	ホストで使用する環境（Ａ工場環境）と、地方工場を識別するコードを同時に表現したい	Ａ工場環境で地方基幹工場を識別したい
「管轄コード」と「リンク情報」を対にして取り扱うことによる影響は？	ホスト内の処理で現在は、管轄コードのみを意識して処理を行っているが、今後は新カテゴリである地方工場を識別する必要がある。 →地方工場を識別する処理・運用には何があるか確認する必要あり？？？	Ａ工場環境で識別し、何かをしたい

付表 4:変更要求-関連システム表の影響システム表現パターン

システムを横方向に記述した方が、各要求に対して影響する他システムを表現できる。

			工場	A工場					B工場				
			業務	仕入れ	印刷	受付業務	印刷
			システム	伝送仕入システム	伝送受付システム
依頼内容	<ul style="list-style-type: none"> ・ホストシステムは部品を仕入れた工場（A工場・B工場）を、「管轄コード」にて識別するが、地方工場は、新たに「リンク情報」も付加して識別する。 ・部品の仕入れ処理において、地方工場の場合は、「リンク情報」に「地方工場コード」を採用する。 ・＜地方工場の定義＞ 管轄コード：「0」、リンク情報：[地方工場コードを意味する数値] （管轄コード：0 → A工場、1 → B工場） 												
上位要求	1	新設する地方工場の部品検品システムは、B工場のシステムをベースに構築。今回稼動する地方工場の仕入れは便宜的にホストの既存環境（A工場環境）を利用する。		△									
	理由	従来は、工場別にホストシステムのオンライン環境を作成していた。新たなオンライン環境作成はホストシステムの資源圧迫にもつながる為、今回はA工場の環境を利用する。											
	説明												
変更要求	1.1	現在はA工場（本社工場）とB工場（〇〇工場）のみしかなく、地方工場が存在しない。初の地方工場設立に合わせ、部品検品システムを稼動させたい。また、今後、同様の地方工場を増やしていきたい。											
	理由	地方工場を設立し、全社レベルで、製品供給能力の向上を図る。											
	説明												
変更要求	1.2	地方工場・部品検品システムのもとであるB工場システムではホスト環境（B工場環境）を使用しているが、今回、地方工場の検品情報をホスト投入する際は、既存のA工場と同じホスト環境（A工場環境）に投入したい。		△									
	理由	<ul style="list-style-type: none"> ○地方工場に登録した部品検品情報をホストA工場環境でも使用する為。 ○ホストA工場環境の下記処理で使用したい。 ・仕入れ ・検品 ・報告書印刷 （他にはないか??? → 要確認、要調査） 											
	説明	地方工場が増設された後も、ホストの運用とA工場の運用は基本的に、現行と同じとしたい。											
変更要求	1.3	新たなカテゴリである地方工場について、検品情報を、既存のホストA工場環境に投入するが、今後、他の地方工場が新設された場合、どの地方工場で仕入れられた部品であるかを識別したい（「管轄コード」と「リンク情報」で識別）。		△									
	理由	<ul style="list-style-type: none"> ・A工場のシステムを運用する際、各地方工場ごとに処理内容を分けるため。 （システムで識別の必要がある処理は??? → 要調査） ・管轄コード、リンク情報のA工場部品検品システムでの使用用途は??? → 要調査 ・A工場部品検品システムでは管轄コード、リンク情報を何に使用しているか??? → 要調査 											
	説明												

管轄コード「0」(A工場を使用)とすることから、「B工場」は対象範囲から除外できた

要調査
A工場のシステムに対する調査を依頼し、影響の有無を判定する