

モーフォロジカルアナリシスを適用した潜在顧客の要求分析手法

ジリエ陽子^{†1†2} 本田耕三^{†1} 中川博之^{†3}
田原康之^{†1} 大須賀昭彦^{†1}

本論文では、Web GUI アプリケーション部品ライブラリ開発を題材に、再利用性の高いソフトウェアのための要求分析手法を提案する。短期間での開発を想定し、ライブラリとして必要なベーシックな機能を厳選して要求獲得できるようにする。さらに、モーフォロジカルアナリシスを適用し、明示的な顧客だけでなく、顧客になる可能性のあるターゲット層すなわち潜在顧客の要求仕様を満たす機能の抽出を行う。Web GUI アプリケーションのグリッド部品ライブラリ開発について本提案手法を適用し評価を行った。従来手法や抽出条件を考慮しない単純組合せ手法に比べ、要求獲得分析の有効性を確認した。

A Requirements Analysis Method for potential customers with Morphological Analysis

YOKO GIRIER^{†1†2} KOZO HONDA^{†1} HIROYUKI NAKAGAWA^{†3}
YASUYUKI TAHARA^{†1} AKIHIKO OHSUGA^{†1}

This paper proposes a requirements analysis method for high reusable software on Web GUI application widget library development. Assuming short period of development, we make it possible to acquire requirements carefully selecting basic functions necessary as library. Furthermore, with morphological analysis, we extract functions which satisfy requirements not only for explicit customers, but also for target to become customers potentially, that is potential customers. We evaluated our method applying to web GUI application widget library. We verified our method is more valid compared to existing method or simple method which does not consider extract conditions.

1. はじめに

顧客の必要性や要求を特定する工程として、様々な要求分析手法が研究されている。KAOS 要求分析手法などを用いて、明示的に存在する既存顧客の要求仕様を機能レベルへブレイクダウンすることは可能である。ソフトウェア製品開発では、あるソフトウェアを拡張して再利用するケースが多いため、あらかじめ汎用的な設計を行うことは効果的である。ただし、そのためにはどう拡張し再利用するか事前にある程度顧客仕様の傾向を把握している必要がある。しかし、顧客になる可能性のあるターゲット層すなわち潜在顧客の要求を抽出する手法は十分に研究されていない。本論文では、ライブラリとしての再利用を前提とした、汎用的なソフトウェアのための要求獲得分析手法を提案する。具体的には Web GUI アプリケーション部品ライブラリ開発を題材に検討を行う。例えばクラウドコンピューティングや HTML5 技術の普及により、ビッグデータの表示を前提とした監視制御システムなど社会インフラシステムを Web GUI アプリケーションとして開発することが考えられている。これらの Web GUI アプリケーションでは、グラフ部品やエクセルのような表を表示するグリッド部品などよ

く使われる部品がある。グリッド部品では複数の属性値をもつ項目をグリッド状に表示し、ソート、スクロールなど様々な機能が考えられ、ライブラリとしての提供を検討する。Web GUI アプリケーションのグリッド部品ライブラリとしては、jqGrid[6]、dhtmlxGrid[5]、DataTables[4]など数多くのオープンソースもしくは商用ライブラリが存在する。これらのライブラリの中で一番良さそうなものを利用すればよいだけのように思われるが、一般にオープンソースや商用のライブラリだけで特定の顧客要求を満たす機能がそのまま実現できることはあまりない。例えば、従来 VB ベースの表を多用したシステムを Web アプリケーションに置き換える場合、ユーザはエクセルのような機能（コピー & ペースト機能、Enter キーによる移動など）や使い勝手を期待する。しかし、オープンソースのライブラリにはエクセルのような機能は充実していない。そこで、オープンソースをベースにカスタマイズされたライブラリを開発することが考えられる。ライブラリとして提供する前提であるため、異なるドメインの複数のアプリケーションでライブラリとして利用できるよう、汎用性をもたせる必要がある。どんな製品にも使えるライブラリを考えると、あらゆる機能を盛り込みたいものであるが、ごく当然に、製品開発工数には制限がある。そこで、必要な機能だけ厳選して開発できることが望ましい。しかし、Web GUI アプリケーション部品ライブラリ開発者は一般に個々のドメイン、例えばエネルギー監視制御システムや電力取引システムについて

^{†1} 電気通信大学
The University of Electro-Communications.
^{†2} 株式会社東芝
Toshiba Corporation
^{†3} 大阪大学
Osaka University

は詳細な知識をもっていないケースがほとんどである。かつ、複数のシステムの要求仕様が同時に明確になっていることもまれである。そこで、明示的に確定している最初の顧客より後の潜在顧客についてもある程度予測を立てられることが必要である。潜在顧客の要求をあらかじめ分析しておくことのメリットのひとつは、品質の高いソフトウェアが開発できることである。五月雨式に要求が明確になるたびに実装するよりもテスト工数の削減になる。別のメリットとしては、魅力的な機能が発見できる可能性が高いことである。経験上、特徴のある魅力的な機能というのは、ベンチマークの結果共通性の低い機能に存在する傾向にある。つまり、ベンチマークで共通性の高い機能を盛り込んだだけでは平均的なライブラリはできるが、既存ライブラリと比較して特徴のないライブラリになってしまう可能性がある。もしくは、既存ライブラリをそのまま使えばよいという結果にもなる。また、ベンチマーク結果から各ライブラリにちらばった有効な機能の集合を見つける必要があるが、先入観にとらわれず効果的に探せることが望ましい。このような要件に効果的な手法として、モーフォロジカルアナリシスの適用が期待できる。

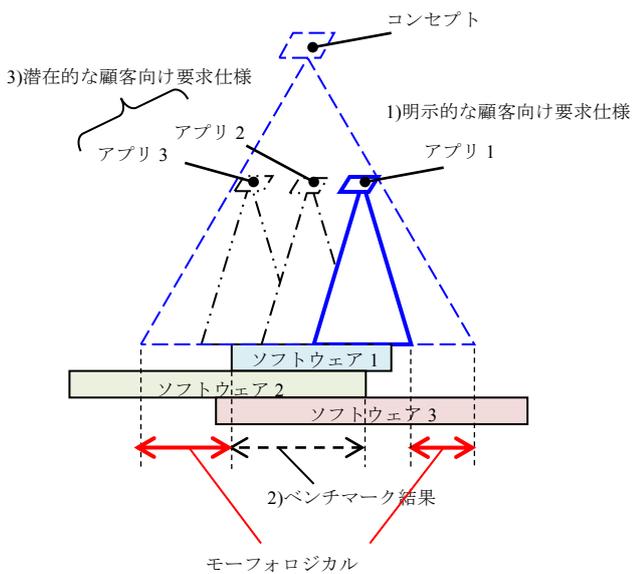


図 1 概念図

Figure 1 Conceptual diagram.

以上より本論文では次の4つのステップで要求獲得分析を行う。

- 1) KAOS 要求分析手法を用いて明示的な顧客の要求仕様を分析する。
- 2) 開発対象のソフトウェアに類似した既存ソフトウェア (Web GUI アプリケーション部品ライブラリ) のベンチマークを行い、共通性の高い一般的な機能を抽出する。
- 3) 潜在的な顧客要求 (以下、Voice of Customer すなわち VoC と記載) を QFD 分析して重要度の高い要求品質をモ

ーフキーとして抽出する。

- 4) モーフォロジカルアナリシスを適用した選択処理を行い、潜在的な顧客の要求仕様を満たす機能を抽出する。モーフォロジカルアナリシスを適用した選択処理では、1) の分析結果である明示的な顧客仕様と 2) ベンチマーク結果の和集合および 3) の分析結果モーフキーを入力とし、重要と思われる機能を絞り込む。最終的な要求仕様は a) 顧客より明示的に示された要求仕様、b) 類似ソフトウェアのベンチマークで共通性の高い一般的な機能を抽出した要求仕様、および c) 顧客より明示的に要求されず、一般的ではないが有効と思われる機能をモーフォロジカルアナリシスで抽出した潜在的な顧客向け要求仕様の和集合である。優先順位は a) が一番高く、次いで b), c) の順に高くなる。以降、本論文では a) を第 1 優先仕様、b) を第 2 優先仕様、c) を第 3 優先仕様とする。

本論文の構成は以下の通りである。第 2 章では、汎用的なソフトウェア開発の課題と要件について述べる。第 3 章では本研究で適用する従来技術について概説する。第 4 章では、本研究提案手法の詳細について述べる。第 5 章では、具体的にグリッド部品ライブラリ開発へ本研究提案手法を適用し、評価と考察を行う。第 6 章では従来研究について述べる。第 7 章では、結論と今後の課題について述べる。

2. 再利用性の高いソフトウェア開発の課題と要件

ソフトウェア製品開発では、一般に工数短縮やノウハウの蓄積といった観点から、あるソフトウェアを別の製品に拡張して再利用することが多い。そこで複数のアプリケーションで再利用できる部分ができるだけ多くなるようソフトウェアの共通部分をあらかじめ抽出できることが望ましい。しかし、共通性を検討できるほど複数のアプリケーションそれぞれの仕様が事前に同時に明確になっていないケースや、開発者がアプリケーション個々のドメインについて詳細な知識をもっていない場合、要求の獲得は困難である。さらに、獲得した要求仕様からどれを再利用性の高い汎用的なソフトウェアの仕様として採用するか検討する必要がある。

本研究の課題としては、次の通りである。

- 1) 再利用性の高い汎用的なソフトウェアの仕様として採用できる要求をいかに獲得するか
- 2) 獲得した要求をもとにいかに汎用的なソフトウェアとしての仕様を作成するか

まず、ソフトウェアを開発するにあたって、開発のトリガーとなるアプリケーション、すなわち明示的に顧客が決まっているアプリケーションの要求仕様を抽出し、機能レベルへブレイクダウンする。要求分析手法としては様々に研究されているが、ソフトウェア開発の目標や、ステークホルダの意図を分析するための要求工学のアプローチとし

て有効であると認識されつつある，ゴール指向要求分析手法を用いる．具体的には，ゴール指向要求分析手法の代表的な方法論である KAOS 要求分析を用いる．特定の顧客向けだけではなく汎用性をもたせるために，共通性の高い仕様を抽出する．具体的には，ベンチマークで共通性の高い一般的な機能を抽出する．さらに，明示的に存在する顧客だけでなく，潜在的に存在する顧客の要求仕様も拾えるようにする．抽出した機能の絞り込みについては，複数の観点から検討する必要がある．このような複雑な問題の解決手法として，モーフォロジカルアナリシスを適用する．

3. 従来技術

3.1 KAOS 要求分析

ゴール指向要求分析はニーズをシステムが達成すべき目標，すなわちゴールととらえ，なぜ・どうやって達成するのか，という観点で分析する要求工学の重要なアプローチである．KAOS はゴール指向要求分析の代表的な分析手法のひとつである．システムゴールを 1) 目標状態，2) システムが達成責任を持つ操作要求と，3) 環境や人が達成責任をもつ仮説の 3 種類に分類し，系統的に分析する．KAOS はゴールモデルを中心とした要求分析を行う．ゴールモデルは，目標とする状態を系統的に分析したモデルである．ゴールを AND/OR リンクと障害リンクにより結合し構造的に表現する．次図 2 は，それぞれ左から AND リンク，OR リンク，障害リンクの例である．

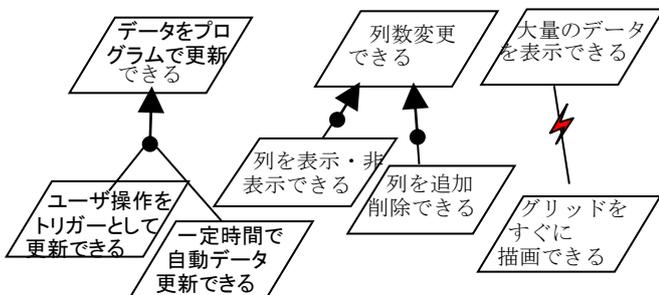


図 2 AND リンク，OR リンクおよび障害リンク

Figure 2 From left to right, AND link, OR link and obstacle link.

3.2 QFD 分析手法

QFD 分析手法は顧客のニーズを技術に結びつけ，どのニーズに答え（製品企画），どの技術を開発するかを決定するための分析手法である．VoC をベースに要求品質を洗い出し，各要求品質を改善する品質特性に変換する．要求品質そのものの重要度と要求品質と品質特性の対応関係により要求品質として重要なものを抽出する．図 3 は要求品質と品質特性をマトリクス表示したものである．各品質特性（例えば B1）は該当する要求品質（A1）に対してどの程度重要かを示す値が格納される．この要求品質に対する品質特性の

重要度と顧客重要度を掛け合わせた数値が得点となる．本論文では潜在顧客の VoC を反映するため，顧客重要度の高い要求品質をモーフキーとして利用する．

		顧客重要度	品質特性				
			B1	B2	B3	B4	B5
要求品質	A1	9	9				
	A2	3		6	6		
	A3	1				3	
	A4	3				3	2
	A5	6					9
得点			81	18	18	12	60

図 3 QFD 分析

Figure 3 Quality function deployment(QFD).

3.3 モーフォロジカルアナリシス

モーフォロジカルアナリシスは，カリフォルニア工科大学の Fritz Zwicky が考案した発想法の一種で「ひとつのアイデアはいくつかの要因が合成されて生まれてきたものである」との考え方をベースとした分析手法である．多次元で定量化できない複雑な問題解決に適した手法で，分割統治手法をベースとしている．分析手順は，まず解決すべき問題をいくつかの構成要素，すなわちモーフキーに分割する．それぞれの構成要素に対する解を具体化する． n 個の要素があれば n 次元の表になる．要素ごとの解を組み合わせ，解決すべき問題の解を導く．ここで，解の組み合わせはある意味ランダムで自由に検討する．ゆえに開発者の主観に影響されない点が，モーフォロジカルアナリシスの長所である．

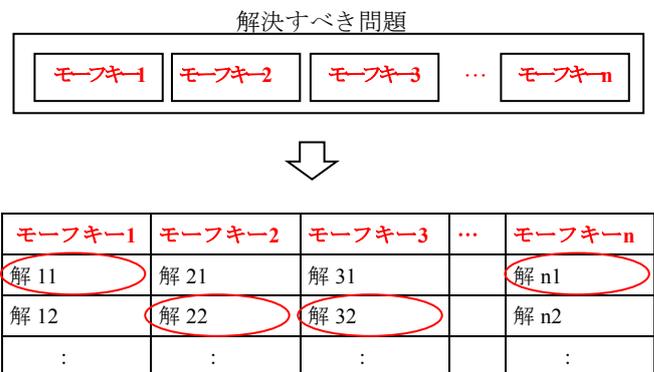


図 4 モーフォロジカルアナリシス

Figure 4 Morphological analysis.

4. 本研究提案手法

4.1 KAOS 要求分析

明示的な顧客があり、開発することが確定しているアプリケーションで使われるライブラリ部分について従来通りトップダウンの要求分析を行う。なお、アプリケーションの機能がそのままライブラリの機能になるのではない点に注意が必要である。すなわち、アプリケーションの仕様のうち、アプリケーション独自ではなく他のアプリケーションでも汎用的に使えそうなのでライブラリに任せる機能について分析する。なお、KAOS 要求分析で抽出された機能が第1 優先仕様に該当する。

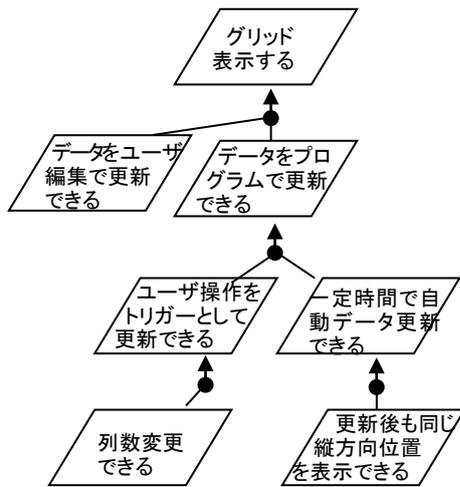


図 5 KAOS 要求分析モデル

Figure 5 KAOS requirements analysis model.

4.2 ベンチマーク

開発対象のソフトウェアに類似した既存ソフトウェアのベンチマークを行う。各ソフトウェアの機能の和集合を作成し、それぞれの機能が各ソフトウェアに存在するかチェックする。ソフトウェアごとに○×をつけ、集計結果として機能ごとにトータルの○の数をカウントし、共通性として記載する。なお、ベンチマーク結果で共通性の高い機能(例えば共通性3や2の機能)が第2 優先仕様に該当する。

表 1 ベンチマーク結果

Table 1 Benchmark results.

機能概要	機能詳細	ベンチマーク結果			共通性
		ソフト1	ソフト2	ソフト3	
A	機能 A1	○	○	○	3
	機能 A2	○	×	○	2
	機能 A3	○	○	○	3
B	機能 B1	×	○	○	2
	機能 B2	○	×	×	1
C	機能 C1	○	×	○	2

4.3 KAOS 要求分析とベンチマークのマージ

KAOS 要求分析の結果とベンチマークの結果をマージした、詳細機能の和集合の表を作成する。なお、この和集合は第2 優先仕様候補に該当する。具体的には、図 5 のような KAOS 要求分析結果のモデルの葉ゴールに対し、表 2 の詳細機能に該当する機能があれば顧客仕様の列を1 とする。ここで葉ゴールとは末端のゴールでその下に別のゴールをもたない、もともと詳細に分析された要求仕様である。該当する機能がない場合、すなわち KAOS 要求分析モデルにのみ存在する機能は詳細機能の行を追加し、共通性の数0、顧客仕様1 とする。顧客仕様の列は、1 以外の行はすべて0 である。なお、これらの分析結果のうち詳細機能と詳細機能の属性は後のステップ 4.5 で利用する。

表 2 第2 優先仕様候補

Table 2 Candidates of the second preferred specification.

機能概要	詳細機能	ベンチマーク結果			詳細機能の属性	
		ソフト1	...	ソフト4	共通性	顧客仕様
		○		○	4	1
A	機能 A1	○		○	4	1
	機能 A2	○		○	3	1
	機能 A3	○		○	3	0
B	機能 B1	×		○	2	1
	機能 B2	○		×	3	0
C	機能 C1	○		○	2	0
D	機能 D1	×		×	0	1
	機能 D2	×		×	0	1

4.4 QFD 分析によるモーフキー抽出

潜在的な顧客の要求仕様を満たす機能を、モーフォロジカルアナリシスを適用して抽出する。まず、QFD 分析によりモーフキーの抽出を行う。VoC をベースに要求品質を洗い出し、品質特性に変換する。要求品質そのものの重要度と要求品質と品質特性の対応関係により要求品質として重要なものをモーフキーとして抽出する。

4.5 モーフォロジカルアナリシスを適用した選択処理

モーフキーに対する解を具体化する。解の候補として 4.3 で作成した第2 優先仕様候補と、その属性として設定した共通性および顧客仕様を(表 2 参照)を利用する。図 6 のように、各モーフキーに第2 優先仕様候補の詳細機能を列挙する。次に、列挙した詳細機能から、まずは機能概要レベルで重要ではないと思われるものを削除する。さらに、詳細機能でも不要なものがあれば削除する。結果が図 7 である。

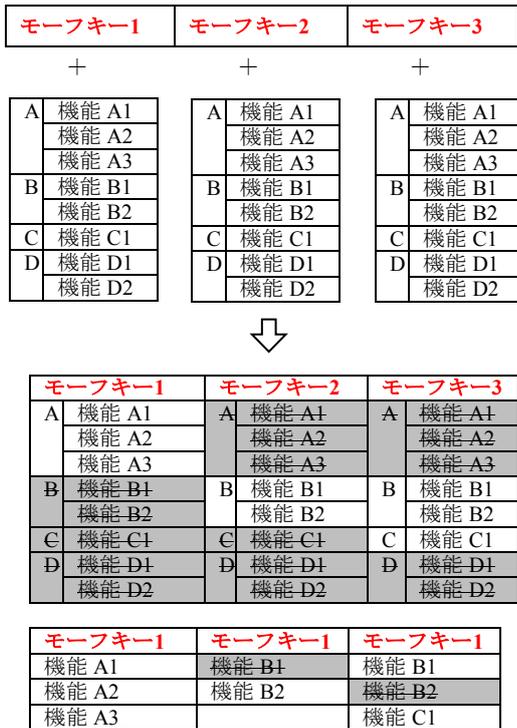


図 6 モーフキーごとの解の間引き
 Figure 6 Thinning solutions according to morph key.

モーフキー1	モーフキー1	モーフキー1
機能 A1	機能 B2	機能 B1
機能 A2		機能 C1
機能 A3		

図 7 モーフキーごとに抽出した解(詳細機能)
 Figure 7 Solutions according to morph key.

次に、図 7 モーフキーごとに抽出した解（詳細機能）と、共通性および顧客仕様を入力として詳細機能を選択する。ここで、図 7 はモーフキー3 つでシンプルな例(詳細機能の組み合わせは $3 \times 1 \times 2 = 6$ 個)であるが、一般に n 個のモーフキーがあれば詳細機能の組み合わせは n 次元の表になる。例えば、モーフキーが 5 個、モーフキーごとに詳細機能が 10 個ある場合、組み合わせの個数は 10^5 個である。そこで、各機能を指標に従い組み合わせを抽出する。明示的な顧客の仕様としては合致しない、ベンチマーク結果で共通性の多いベーシックな機能にもあてはまらない、すなわち従来手法では抽出しにくい、潜在顧客の要求仕様にマッチしている機能を拾うため、指標としては次の条件に着目する。

- ・顧客より明示的に要求仕様として示されない
- ・既存のライブラリにあまり存在しない
- ・VoC より設置した目標値(=モーフキー)を改善する
- ・モーフキーごとの代替機能が少ない

図 8 は選択処理のフローチャートである。間引き処理の出力であるモーフキーごとに抽出した詳細機能に対し、第 2 優先仕様候補作成時に付加した共通性および顧客仕様の属

性を結び付ける。モーフキーごとに抽出した詳細機能の一つずつ組み合わせる。組み合わせごとに、抽出条件にマッチするかチェックする。図 9 は組み合わせを抽出する例である。モーフキーごとに機能を組み合わせ、(カッコ)の中の共通性と顧客仕様の数を合計する。例では、抽出条件：共通性合計 5 以下かつ要求仕様合計 1 個以下である。

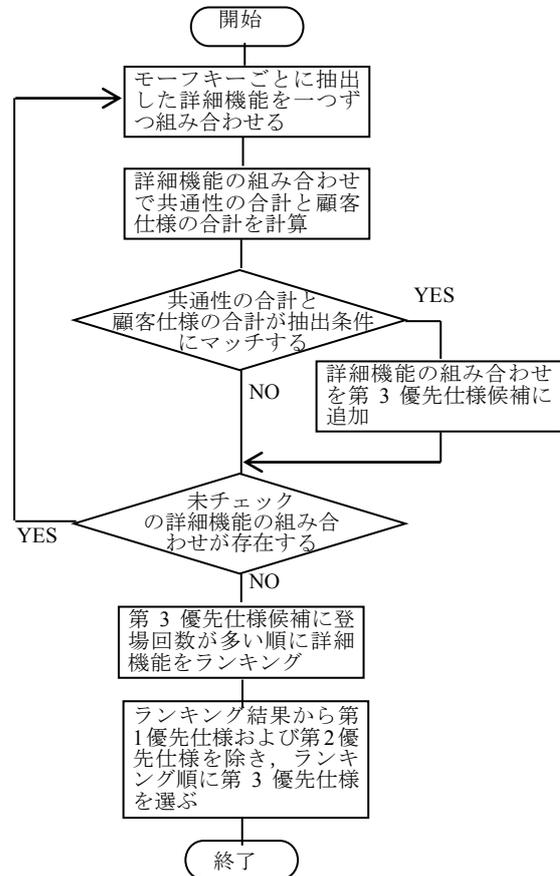


図 8 フローチャート
 Figure 8 Flowchart.

モーフキー1	モーフキー1	モーフキー1
機能 A1(0,1)	機能 B2(3,0)	機能 B1(1,0)
機能 A2(2,0)		機能 C1(1,1)
機能 A3(1,0)		

※カッコは(共通性の数, 顧客仕様の数)

組み合わせを作成 ↓

モーフキー1	モーフキー2	モーフキー3	合計
機能 A1(0,1)	機能 B2(3,0)	機能 B1(1,0)	(4,1)
機能 A1(0,1)	機能 B2(3,0)	機能 C1(1,1)	(4,2)
機能 A2(2,0)	機能 B2(3,0)	機能 B1(1,0)	(6,0)
機能 A2(2,0)	機能 B2(3,0)	機能 C1(1,1)	(6,1)
機能 A3(1,0)	機能 B2(3,0)	機能 B1(1,0)	(5,0)
機能 A3(1,0)	機能 B2(3,0)	機能 C1(1,1)	(5,1)

※抽出条件の例：共通性合計 5 以下かつ要求仕様合計 1 個以下

選択された組み合わせ
選択されなかった組み合わせ

図 9 抽出した組み合わせ
 Figure 9 Extracted combination.

条件式は、モーフキーの個数 n 個、モーフキー k の共通性を C_k 、顧客仕様 V_k とすると、

$$\sum_{k=1}^n C_k < P \quad \text{かつ} \quad \sum_{k=1}^n V_k < Q$$

ここで、具体的な閾値 P および Q は、一般に工数に上限があるため、抽出された機能の数で調整する。経験上それぞれの最大値の 5 割程度が妥当である。図 7 でモーフキーごとに抽出した解のテーブルに登場する機能(A1, A2, A3, B1, B2, C1)が、図 9 の解の組み合わせに登場する回数をランキングする。なお、図 9 の組み合わせのうち、共通性の合計と顧客仕様の合計が抽出条件を満たし、選択された組み合わせ(白い行)のみ利用する。図 10 ランキング結果から顧客仕様と共通性の多い仕様をまず削除する。残った機能で、ランキング順に A3, B1, C3 の順に選ばれる。なお、機能をいくつまで選ぶかは工数に依存して調整する。

機能 A1(0,1)	4
機能 A2(2,0)	0
機能 A3(1,0)	2
機能 B1(1,0)	2
機能 B2(3,0)	3
機能 C1(2,0)	1

ランキング結果から顧客仕様=1 もしくは共通性 ≥ 3 を削除した例

図 10 ランキング結果
 Figure 10 Ranking results.

以上のように、1)明示的に示された顧客仕様、2) ベンチマークで共通性の高い仕様、3) モーフオロジカルアナリシスで抽出した機能の和集合が最終的なライブラリの仕様となる。優先順位は 1), 2), 3)の順に高くなる。

5. Web GUI アプリケーションのグリッド部品ライブラリ開発への適用と評価

本研究手法を Web GUI アプリケーションのグリッド部品ライブラリの仕様作成に適用する。

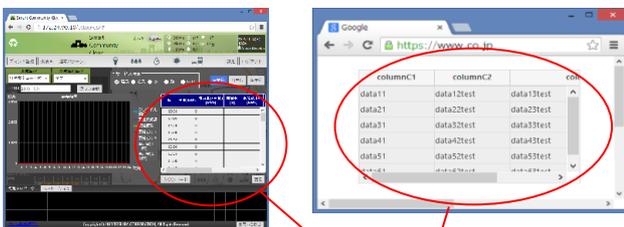


図 11 グリッド部品
 Figure 11 Grid widget.

まず、明示的な顧客向けアプリケーションとしてエネルギー監視制御システム Web GUI アプリケーションの要求分析を行う。さらに、他システムの Web GUI アプリケーションの仕様と比較し、本研究の有効性を検証する。

エネルギー監視制御システム Web GUI アプリケーションの要求分析を行う。さらに、他システムの Web GUI アプリケーションの仕様と比較し、本研究の有効性を検証する。

5.1 KAOS 要求分析

エネルギー監視制御システム Web GUI アプリケーションの要求仕様のうち、グリッド部品に任せる仕様に着目した KAOS 要求分析のゴールモデルを作成した。“グリッド表示できる”というゴールをルートとして、システム画面 3 つ分について分析したところ、画面一つあたり 20 前後になった。3 つ分のモデルをマージした結果、ゴール数 28、抽出できた機能数 11 個となった。

5.2 ベンチマーク

既存のグリッド部品ライブラリの類似ソフトウェアとして、オープンソースの jqGrid[6]と DataTables[4]、商用の dhtmlxGrid[5]、マイクロソフトエクセルを対象としてベンチマークを行った。なお、エクセルは Web GUI アプリケーション向けグリッド部品ライブラリではないが、一般に広く使われているためベンチマーク対象とした。ただし、エクセルの仕様は膨大であるため、まず jqGrid, dhtmlx, DataTables のベンチマークにより詳細仕様の和集合を作成し、この和集合に存在する仕様についてのみエクセル機能の確認を行った。洗い出した詳細機能数について以下表 3 に示す。和集合の合計数は 314 だった。

表 3 詳細機能数

Table 3 The number of detailed functions.

機能概要	詳細機能数
グリッド操作	29
データ読込	23
行操作	42
列操作	38
セル操作	51
選択	11
ソート	9
フィルタリング	8
グルーピング	7
ドラッグ&ドロップ	10
クリップボード操作	12
クッキー操作	16
データ保存	18
サブグリッド表示	5
ツリーグリッド表示	15
簡単操作	17
ナビゲート機能	3
合計	314

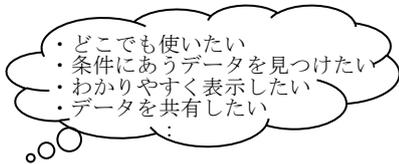
5.3 KAOS 要求分析とベンチマークのマージ

KAOS 要求分析の結果とベンチマーク分析の結果をマージしたところ、抽出した機能数 11 個のうちベンチマーク結果に存在する機能 9 個、ベンチマーク結果に存在しない機

能 2 個となった。

5.4 QFD 分析によるモーフキー抽出

QFD 分析によりモーフキーの抽出を行った。潜在顧客の声として、実際に社内での要望を集めたり、一般に Web GUI アプリケーションのグリッド部品に期待されている要求を参考にした。



要求品質	品質特性	改善方向
モバイル	描画時間, ショートカットキー	↑
検索	ソートの種類(テキスト, 数値), ...	↑
解析	範囲選択, ...	↑
可視化	色変え, 階層表示, ツールチップ, ...	↑
共有	データ保存, 印刷時の見栄え, ...	↑

図 12 要求品質と品質特性

Figure 12 Required quality and quality characteristics.

5.5 モーフオロジカルアナリシスを適用した選択処理

まず、モーフキーごとに 4.5 の手順でモーフキーごとの解を抽出する。次表 4 の 1 行目はモーフキー、2 行目以降は機能概要名、(かっこ)内の数値は間引きした後の詳細機能数である。

表 4 モーフキーごとに間引きした後の詳細機能数

Table 4 Thinned functions according to morph key.

モバイル	検索	共有	解析	可視化
ソート(2)	ソート(5)	データ保存(6)	ソート(5)	セル操作(2)
フィルタリング(1)	フィルタリング(2)		フィルタリング(2)	ソート(2)
ドラッグ&ドロップ(2)			グループピング(4)	フィルタリング(2)
簡単操作(3)			ドラッグ&ドロップ(6)	グループピング(2)
			サブグリッド表示(2)	サブグリッド表示(2)
			ツリーグリッド表示(3)	ツリーグリッド表示(3)

表 5 はモーフキーごとに間引きした後の詳細機能のグループから第 3 優先仕様(本提案の列)の抽出結果と各手法の比較である。なお、表 5 の算出については詳細機能名、共通性、顧客仕様かどうかを入力として図 10 ランキング結果を出力する簡単な計算プログラムを作成し、自動計算を行っている。

表 5 分析結果

Table 5 Analysis results.

機能概要	詳細機能	従来手法	単純組合せ	本提案	共通性	顧客仕様
操作セル	セルテキストスタイル設定	○		○	4	0
	セル背景色設定	○		○	4	0
ソート	ソートタイプ設定	○	○	○	4	1
	ユーザカスタマイズソート	○	○	○	4	0
	行ソート	○	○	○	4	1
	ソート前の状態へ復帰		○	○	1	0
	マルチソート		○	○	3	0
ゲタフィル	全グリッドフィルタリング		○	○	1	0
	個別フィルタリング	○	○	○	4	0
グループピング	全グループピング解除	○		○	4	0
	指定グループピング解除	○		○	4	0
	全グループピング展開				2	0
	指定グループピング展開	○		○	4	0
ドラッグ&ドロップ (以下 D&D と略)	D&D 有効(行)	○	○	○	4	0
	D&D 有効(列)	○	○	○	4	
	D&D 順/デフォルト順切り替え(行)				1	
	グリッドからグリッドへの行データ D&D				2	
	グリッドからツリー要素への D&D				1	
	ツリー要素からグリッドへの D&D				1	
データ保存	XML データ出力		○		3	
	CSV データ出力		○		3	
	行追加削除情報の保存設定		○	○	2	
	変更行のみ保存		○	○	2	
	Excel 出力		○		3	
	PDF 出力		○		3	
ツグサドリブ	行展開			○	2	
	行クローズ			○	2	
ドグツリリ	指定ノードクローズ			○	2	
	ノード展開			○	2	
	ソートツリー			○	2	
作簡単操	やり直す			○	2	
	元に戻す			○	2	
	TAB 移動順設定			○	2	

表 5 の従来手法で選ばれた仕様とは、ベンチマーク結果に

より共通性が高いと判定された仕様しくは顧客より明示的に必要とされた仕様である。単純組合せ手法で選ばれた仕様は、基本的には図 8 のフローチャート同様の手順で、抽出条件による選択をせずに選ばれた仕様である。なお、最終的な仕様は表 5 に含まれるものだけではなく、第 1 優先仕様、第 2 優先仕様および第 3 優先仕様の和集合である。次に、従来手法、単純組合せ手法および本提案手法の比較を行う。従来手法では、共通性の低い仕様は選ばれない。つまり、どのソフトウェアにも存在するような、重要性は高いがある意味、ありきたりな機能のみが選ばれる傾向にある。単純組合せ手法では、モーフキーごとの詳細機能数(表 4 参照)が少ない機能から選ばれる。つまり、モーフキー「検索」や「共有」に結びついた詳細機能は選ばれるが、「解析」や「可視化」などモーフキーあたりの機能数が多いものは選ばれない。これは、組み合わせで登場回数が多いものを選ぶためである。つまり、単純組合せでは解析するまでもなく、モーフキーごとの詳細機能数が少ないものが選ばれる結果となる。本提案手法は、ベンチマークで共通性の高いものや顧客仕様は当然に採用しつつ、抽出条件によって選択しており、4.5 に記載の指標を満たす結果となっている。つまり、明示的に顧客から示されていないが VoC より設置した目標値を改善し、代替機能はあまりない機能までも抽出できる結果となっている。

また、新たに試作予定となっているエネルギー売買システムで本提案手法の評価を行った。エネルギー売買システムをグリッド機能に限定して KAOS ツリーに分析したところ、ゴール数 20、抽出できた詳細機能数 12 個、うちベンチマークを含むこれまで洗い出した詳細機能(第 2 優先仕様候補)に存在する機能 7 個、存在しない機能 5 個となった。表 5 の第 3 優先仕様と一致する新システムの第 1 優先仕様の個数を手法ごとに比較したところ、次のように、本提案手法が一番一致する結果となった。

表 6 試作予定システムにおける比較
 Table 6 Comparison on new prototype system.

	従来手法	単純組合せ	本提案
一致した項目数	4	4	6
一致しなかった項目数	3	3	1

6. 従来研究

効率的に要求獲得する手法について、久代ら[2]の多次元ヒアリング手法を用いた研究があるが、コンセプト形成のための設計会議に関するものであり、再利用性の高いライブラリソフトウェアの具体的な要求仕様獲得には向いていない。Web GUI アプリケーションの限られた期間と工数で

の開発については、三部ら[3]の研究があるが、プロトタイプングを用いており特定のアプリケーション開発にはよいが、ライブラリとして複数のアプリケーションで共通で使われるソフトウェアには向いていない。類似既存システムの情報を用いた研究については、海谷ら[1]の研究があり、本論文では要求獲得者が不案内な分野で要求獲得を行うという状況で既存ソフトウェアの情報を利用する点は同様であるが、既存にはないようで、かつ潜在顧客の要求仕様にマッチした機能の追加についてまでは考慮されていない。また、Web GUI アプリケーションのグリッド部品ライブラリとしてすでに jqGrid[6]や dhtmlxGrid[5]など存在するが、汎用的である反面プリミティブなくつかの関数を組み合わせると一つのまとまった機能を実現するようになっていたり、API の利用に制限があったりなど使い勝手に向上の余地がある状況である。また、jqGrid[6]や dhtmlxGrid[5]など各ライブラリの和集合としてはそれなりに機能がそろっているが、個々のライブラリとしては機能が足りない状況にある。すなわち、有効な機能を集約した一つのライブラリというものはなく、複数のライブラリに有効な機能が分散している傾向にある。

7. おわりに

本論文では、要求獲得者が不案内な分野での再利用性の高いソフトウェア開発のための要求仕様作成を支援する手法を提案した。従来通りの要求分析に加え、開発対象のソフトウェアに類似したソフトウェアのベンチマークを行い共通性の高い機能を洗い出した。さらにモーフォロジカルアナリシスを適用し、潜在顧客の要求分析を行った。実際に Web GUI アプリケーションのグリッド部品ライブラリ開発に適用し、本提案手法の有効性を検証した。今後の課題としてはモーフキー抽出時に策定した品質特性を活かした分析を行うことや、複数のシステムへ本手法を適用し、有効性のさらなる検証である。

参考文献

- 1) 海谷治彦, 北澤直幸, 長田晃, 海尻 賢二: 類似既存システムの情報を利用した要求獲得支援システムの開発と評価, 電子情報通信学会論文誌, Vol.J93-D, No.10, pp.1836-1850 (2010).
- 2) 久代紀之, 大澤幸生: 多次元ヒアリングと階層的な要求統合プロセスによる要求獲得手法, 情報処理学会論文誌, Vol.47, No.10, pp.2909-2916 (2006).
- 3) 三部良太, 河合克己, 竹内拓也, 石川貞裕, 福土有二: Web アプリケーションのユースケース駆動プロトタイプによる要求獲得方法, 情報処理学会論文誌, Vol.49, No.4, pp.1669-1679 (2008).
- 4) DataTables
<https://datatables.net/>
- 5) dhtmlxGrid
<http://dhtmlx.com/docs/products/dhtmlxGrid/>
- 6) jqGrid
<http://jqgrid.com/>