

エンドユーザ主導開発のためのビジネスロジックの定義方式の提案

中所 武司[†]

[†] 明治大学理工学部情報科学科 〒214-0033 神奈川県川崎市多摩区東三田 1-1-1

E-mail: [†] chusho@cs.meiji.ac.jp

あらまし 変化の激しい時代には、業務の知識を有するエンドユーザ主導のアプリケーション開発とその保守が重要になると考え、その技法を研究してきた。一方、持続可能な社会実現の一環として、IT技術の適用により資源の節約や環境保全を達成することが期待されている。そこで、日常生活の中でのエコ活動に有用なシステムをその担当者自身が開発・運用できれば大きな効果が期待できるという観点から、この分野を例題として、3層アーキテクチャを前提としたビジネスロジックの定義方式を提案する。エンドユーザには、ユーザインタフェース (UI) を起点とした業務処理内容の理解が容易と思われるので、ビジネスロジック処理 (BL) やデータベース処理 (DB) を {UI→BL→DB→BL→UI} という一連のワークフローの中に位置づけて定義するテンプレートを使用する方式についてケーススタディを実施し、その効果を確認した。

キーワード エンドユーザコンピューティング、ビジネスロジック、3層アーキテクチャ、テンプレート

Definitions of Business Logic for End-User-Initiative Development

Takeshi CHUSHO[†]

[†] Department of Computer Science, School of Science and Technology, Meiji University, Kawasaki, 214-0033, Japan.

E-mail: [†] chusho@cs.meiji.ac.jp

Abstract The development of Web applications should be supported by business professionals themselves since Web applications must be modified frequently based on their needs. In our recent studies with the three-tier architecture of user interface, business logic and database, the definition method of the business logic is one of key technologies. This paper describes how the business logic is expressed. For end-user-initiative development, the business logic is defined by using the template based on the UI-driven approach. It is confirmed that the template is useful for requirement specifications.

Keyword End-user computing, Business Logic, Three-Tier Architecture, Template

1. はじめに

インターネット上で Web アプリケーションが普及し、クラウドコンピューティングが注目されるなど、ソフトウェアのサービス化が促進されている。

我々は、変化の激しい時代には、エンドユーザ主導のアプリケーション開発とその保守が重要になるという観点から、10年来の研究を行ってきた。特に、小さな部門や個人の業務を対象とする中小規模の Web アプリケーションに関して、低コストで短期間に開発するとともに、頻繁な機能変更を伴う保守にも対応するために、その分野の業務の専門家主導で開発・保守ができるような技法を研究してきた [2]。

1980年代以降エンドユーザコンピューティングが注目され、その定義、分類、管理に関する研究[1,6,12]が行われてきたが、最近では、エンドユーザ主導開発に関する文献[7,8]も散見される。業務の知識を有するエンドユーザが自らの業務を代行するシステムを自ら開発するためには、はじめにその要求仕様を明確に定義

する必要がある。

エンドユーザになじみのある Web ブラウザをユーザインタフェースとする Web アプリケーションに関して、我々の過去の研究では、コンポーネントベースの技術としてのアプリケーションフレームワークとビジュアルモデリング技術を用いて、ユーザインタフェースおよび比較的簡単なデータベースを構築する方法を実現してきた。

しかしながら、ビジネスロジックに関しては、多種多様なものが存在するため、同様の方法では必ずしもエンドユーザによる実装は容易ではない。そこで、本報告では、最初に例題アプリケーションに関して、多種多様なビジネスロジックの分類を通して、その定義方法を探る。次にその結果に基づいて、サービス提供側の視点に立ったビジネスロジック定義のテンプレートを導入し、その適用実験としてのケーススタディを実施する。

2. 基本的アプローチ

本研究の基本的なアプローチを図1に示す。ビジネスレベルで、エンドユーザ（業務の専門家）はビジネスモデルを提案する。サービスレベルで、ドメインモデルが作成され、必要なサービスが決められる。ソフトウェアレベルでは、コンポーネントを用いてドメインモデルが実装される。

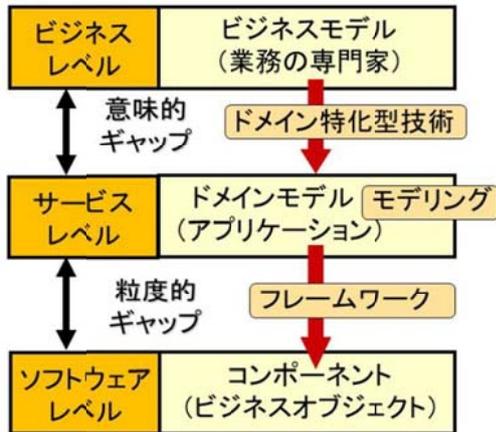


図1. エンドユーザ主導開発のアプローチ

本アプローチでは、コンポーネントとドメインモデルの間の粒度的ギャップは、ビジネスオブジェクトやデザインパターンやフレームワークで補われる。一方、サービスレベルとビジネスレベルの意味的ギャップは、ドメイン特化型技術[11]で補われる。

3層アーキテクチャを前提としたエンドユーザ主導開発では、最初にユーザインタフェース（UI）、ビジネスロジック（BL）、データベース（DB）のどれに注目するかにより3種類のアプローチが考えられるが、DB設計はエンドユーザには難しく、UIまたはBLの視点からのアプローチが適切と思われる。

特にエンドユーザにわかりやすいUI駆動型アプローチでは、フォーム定義を伴うフレームワークを研究してきた。アプリケーション固有のビジネスロジックはフォーム定義に含むとともに、特定分野共通のビジネスロジックはフレームワークにあらかじめ組み込んでおく。しかし、このアプローチは、ワークフローやDB主体のバックエンドシステムには向かない。

このようなワークフロー主体のバックエンドシステムに対しては、モデル駆動型のアプローチが適しており、フォームおよびフォーム変換を定義すればソースプログラムを自動生成するようなビジュアルモデリングツールが有効である。

これらの研究を通じてのエンドユーザ主導開発の主要な課題として、多様なビジネスロジックの定義方法があげられる。これまで、スクリプト言語の開発、ルール表現の導入などを試みてきたが、プログラミン

グの概念は必要である。最近では、特定分野対応のタイルテンプレートをあらかじめ用意し、エンドユーザはこれらのテンプレートを組み合わせて業務処理を記述する方法で問題の解決を図ったが、十分なテンプレートの集合を用意するという課題はある。そこで、本報告では、まずビジネスロジックの分析を行う。

3. ケーススタディのための例題

3.1. 再利用促進サービス

ビジネスロジック分析の例題アプリケーションとして、不用品の再利用促進サービスを取り上げる。昨今、情報技術（IT）を応用して持続可能な社会のための環境保護に貢献する（Green-by-IT）が期待されており、地方自治体が運営する地域住民のための不用品交換サービスやローカルに運営されている不用品再利用のための中古物品の販売店に関して、もし、これらの担当者自身がWebアプリケーションを開発してWebサイトを立ち上げることができれば、扱える品物の量や利用者の数は格段に増加すると思われる。

さらに、このサービスに関しては、多種多様なビジネスロジックが存在するので、その分析のためのケーススタディとしても適していると言える。

3.2. 再利用促進サービスの調査

再利用促進サービスの現状を調べるために、インターネット検索を実施[3]した。その結果、以下のようなことが判明した。

- ・多くの地方自治体が環境問題への対策の一環として、不用品の再利用支援活動をしているが、インターネット上ではその活動内容を紹介しているだけで、実際のサービスは自治体のカウンタで実施している。

- ・本サービスに関して多種多様な規則があり、その内容は個々の自治体ごとに異なる。

例えば我々のキャンパスのある川崎市では、市民が不用品を持ち寄るフリーマーケットを主催している。東京23区の幾つかの区役所でも類似のイベントを実施している。家庭内の不用品を預かり、欲しい人に仲介するサービスもあるが、いずれも自治体のカウンタでの受付処理が主で、インターネットによる運営はしていない。

3.3. ケーススタディによるビジネスロジックの分析

エンドユーザによるビジネスロジックの定義を支援し、その実装を自動化するために、インターネット検索で収集したビジネスロジックを分類することにした。分類に関しては、一般性があると思われる文献[13]の以下のような5種類のビジネスルールのカテゴリを用いることにした。

(1) **事実(Facts)**: 業務上の常に真である事柄あるいはその関連/関係であり、データ実体や属性に関する

不変のもの

(2) **制約(Constrains)**: システムまたはユーザのアクションを制限するもので、「しなければならない」「してはならない」「だけ」などの表現を含むものなど

(3) **行為(Action Enablers)**: 人やシステムに対して特定の条件下でアクションを引き起こすもので、「もし～ならば～する」という表現のものなど

(4) **推論(Inferences)**: 推論知識ともよばれ、ある条件下で成立する新たな事実を導くもので、「もし～ならば～となる／～とみなされる」という表現のものなど

(5) **計算(Computations)**: 特定の数式やアルゴリズムをコンピュータが実行する計算

なお、ビジネスロジックはビジネスルールとワークフローからなるという定義[9]もあるが、本報告では、今回の例題に関して、ビジネスルールとビジネスロジックをほぼ同じ意味で使用する。

4. 要求定義としてのビジネスロジックの分類

4.1. 5種類のカテゴリ

本節では、インターネット検索で収集した多種多様なビジネスロジック（業務規則）を5種類のカテゴリ〔事実、制約、行為、推論、計算〕に分類するというケーススタディの結果について述べる。

4.2. 提供者と引取者の資格と身元確認

ほとんどの自治体では、提供者と引取者に関する資格制限と身元確認を実施しているので、以下の3例の業務規則を取り上げる。

(A1)市内の住民または市内に勤務している18歳以上の人は登録できます

(A2)業者は登録できません

(A3)身元確認が必要です

規則(A1)は制約と行為のいずれかに分類されると思われるが、以下のように言い換えれば制約であろう。

(A11)申請者は、市内在住または市内に勤務しており、かつ18歳以上の場合にのみ登録される

一方、以下のように言い換えれば行為となる。

(A12)もし、申請者が市内在住または市内に勤務しており、かつ18歳以上ならば、登録する

これは2種類のカテゴリに分類可能な典型的な例で、(A1)と(A11)はサービス要求者の視点からの表現、(A12)はサービス提供者あるいはソフトウェア要求仕様(SRS: software requirement specifications)の視点からの表現となっている。

(A2)は制約であるが、(A3)は、先の(A1)と同じように、以下のような言い換えにより制約にも行為にも分類される。

(A31)申請者は、登録に際して身元確認できるものを提示しなければならない

(A32)もし申請者が登録希望の時は身元確認する

4.3. 対象物品の登録

物品登録に関する規則は多いが、ここでは代表例として以下の3件を取り上げる。

(B1)登録物品は家庭内で使用されていたものに限る

(B2)大きな家具や高価な宝石などの展示できないものも登録できるが、預からないで自宅保管のこと

最初の規則(B1)は、以下のように言い換えることにより、事実、制約、行為のいずれかに分類される。

(B11)登録物品は家庭内で使用されていたもの

(B12)家庭内で使用されていた物品のみ登録できる

(B13)もし家庭内で使用されていた物品ならば登録

この中で、(B1)、(B11)、(B12)はサービス要求者の視点からの表現、(B13)はサービス提供者あるいはSRSの視点からの表現となっているが、本サービスがWebサイトでの運用に限られるならば、以下のような規則になる。

(B14)システムは提供者に対し、登録前に、物品が家庭内で使用されていたものであるという誓約を求める

規則(B2)も、(B1)と同様に言い換えることにより、事実、制約、行為のいずれかに分類される。ただし、Webサイトでの運用を前提にすれば、すべての物品は提供者の自宅で保管され、この規則は不要となる。

4.4. 登録できない物品リスト

登録できない物品に関する規則も必ず明記されている。典型的な例は次のようなものである。

(C1)以下の物品は登録できません

そのような物品の例を以下に示す。

(c1) 破損しているもの

(c2) 高額品（3,000円より高額の値をつけるもの）

(c3) 動植物（植物、動物、昆虫等）

(c4) 品質の安全性確認、維持管理等が困難なもの

(c5) 特別の安全管理を必要とするもの（危険物等）

(c6) 飲食物、化粧品、薬品類、アルコール類など

(c7) 法律で販売・取引を禁止されているもの

(c8) 仏壇などの宗教上のもの

(c9) 管理者の判断で不適当とみなしたもの

この規則は、以下のように言い換えることにより、事実、制約、行為のいずれかに分類される。

(C11)リストに含まれる物品は登録されません

(C12)リストに含まれる物品は登録できません

(C13)もし、登録希望の物品がリストに含まれるならば、登録を断る

なお、Webサイトでの運用を前提にすべての物品を提供者の自宅で保管ならば、次の規則になる。

(C14)システムは提供者に対し、登録前に物品が登録できない物品リストに含まれないという誓約を求めるさらに次のような規則が付加されるかもしれない。

(C15)登録された物品は、管理者のチェックののちに
掲示される

4.5. 登録手順

物品の登録手順に関する規則の例を以下に示す。

(D1)提供者は登録時に身元確認できるものの提示を
求められる

(D2)提供者は5件まで登録できます

(D3)1件あたりの登録料は200円です

最初の2つの規則は制約で、最後の規則は事実と考
えられが、(D2)と(D3)は以下のように言い換えること
で、行為カテゴリにもなりうる。

(D21)もし、提供者が物品登録希望ならば、その提供
者が5個以上の登録をしていないことを確認する

(D31)もし、提供者が物品登録希望ならば、1件当
たり200円を徴収する

4.6. 登録物品の属性

登録物品の属性に関する多種多様な規則があるの
で、いくつかの例を示す。

(E1)登録物品の写真を貼付できる

(E2)貼付できる写真は3枚までです

(E3)人物が写っている写真は貼付できません

(E4)無料の物品と有料の物品がある

(E5)物品につけられる価格の上限は5000円です

最初の規則(E1)は事実、(E2)、(E3)の規則は制約に属
するが、次のような言い換えにより行為カテゴリにも
なる。

(E21)もし、写真を貼付するならば、写真の数が3枚
以下であることを確認する。

(E31)もし、写真を貼付するならば、写真の中に人
物が写っていないことを確認する。

一方、(E4)、(E5)の規則は事実とみなせるが、(E5)
については、以下の言い換えにより制約あるいは行為
にもなる。

(E51)提供者は、物品に5000円までの価格をつけら
れる

(E52)もし、物品に価格がつけられたならば、その
額が5000円以下であることを確認する

4.7. 引取者の決定

引取者を決定する規則として、以下の例を示す。

(F1)物品の引取希望者自身が提供者と交渉するた
めに、提供者の情報は引取希望者に知らされる

(F2)引取希望者の数が登録物品の数よりも多い場
合は引取者を抽選で決める

(F3)引取希望者は、抽選対象の物品について2件
まで応募できる

(F4)もし品物の故障や事故があった場合には、双方
(提供者と引取者)で処理していただきます

最初の規則(F1)は事実、(F2)、(F3)は制約、(F4)は行

為とみなせる。最初の3つの規則も以下のように言い
換えれば、行為とみなせる。

(F11)もし、引取希望者がいるならば、物品の提供
者の情報を引取希望者に知らせる。

(F21)もし、引取希望者の数が登録物品の数よりも
多いならば、引取者を抽選で決める。

(F31)もし、引取希望者が抽選対象の物品に応募
するならば、その引取希望者がすでに2件以上応募して
いないことを確認する。

4.8. 登録物品の削除

登録物品を削除する規則として、以下の例を示す。

(G1)登録後5週間経過して残っている物品は、提供
者に返却される

(G2)展示期間は登録後3か月です

いずれも事実とみなせるが、たとえば(G1)の場合
は次の言い換えによって行為あるいは推論とみなせる。

(G11)もし、登録物品が5週間経過して残っている
ならば、提供者に返却する

(G12)もし、登録物品が5週間以内に売れなかった
ならば、その物品は提供者に返却されるべき物品である。

4.9. ケーススタディの考察

以上のように、インターネット検索で収集した不
用品再利用促進サービスを実施している地方自治体の運
用規則を{事実、制約、行為、推論、計算}の5種類
のカテゴリに分類してみた。計算カテゴリに属するも
のは無かったが、この分類実験で、ビジネスルールの
記述の視点が異なれば属するカテゴリが異なるという
問題が明らかになった。

地方自治体が提供するサービスの運用規則は、お
おむね一般市民(サービス利用者)の立場で記述されて
いるが、Webアプリケーションのソフトウェア要求
仕様書(SRS)あるいはシステム(サービス提供者)
の視点で運用規則を言い換えると、行為カテゴリに属
する表現になることが多い。

今回採用した5種類のカテゴリを提案している文献
[13]では残念ながらその点が不明確になっているので、
今回のケーススタディの結果をエンドユーザ主導開発
におけるビジネスロジック記述方式に生かすことは難
しいと思われる。

なお、文献[10]では、エンタープライズシステムの
要件定義の中に記述されているビジネスルールの分析
結果として、大部分がアクションイネーブラ(本報告
での行為カテゴリ)であったという報告がある。

5. ビジネスロジックの要求定義

5.1. ビジネスロジック定義用テンプレート

慈善事業としてのチャリティショップや中古品を
扱うリサイクルショップなどに有用と思われる再利用

支援システムは、エンドユーザ主導で開発するWebアプリケーションの典型的な例と思われる。この場合の重要課題の一つは、多種多様なビジネスロジックのカスタマイズ方法である。

前節のビジネスロジック分類の結果から、ビジネスロジックはサービス提供者あるいはシステムの視点で統一的に記述することが重要であるといえる。さらに本研究では、図2の上部に示すように、典型的な3層アーキテクチャを前提にしているの、要求仕様定義段階でのビジネスロジックをUI、BL、DBの組み合わせで表現することが考えられる。エンドユーザ主導開発では、特にUI駆動型のアプローチが適しているの、以下のようなテンプレートを導入した。

- ①UI：システムは利用者から要求を受け取る
- ②BL：システムはその要求を処理
- ③DB：システムは必要に応じてDBにアクセス
- ④BL：システムはDBアクセス結果を処理
- ⑤UI：システムは結果を表示

このテンプレートは、図2の下部に示すように、ビジネスロジックの典型的な処理プロセスが{UI→BL→DB→BL→UI}であることを意味する。これは、以下のような住民票を入手するプロセスを想定すれば、エンドユーザにとって理解しやすいと思われる。

- UI：市民は住民票申請書を窓口へ提出
- BL：窓口担当者は申請書をチェックし、後方のシステム操作担当者に渡す
- DB：システム操作担当者は申請書の情報を入力し、住民票を出力する
- BL：窓口担当者は出力内容をチェックする
- UI：窓口担当者は住民票を申請者に渡す

これは、ユースケースのテキスト表現によるシナリオ記述と似ているが、システム側の処理内容をUI、BL、DBに分割して定義している点異なる。

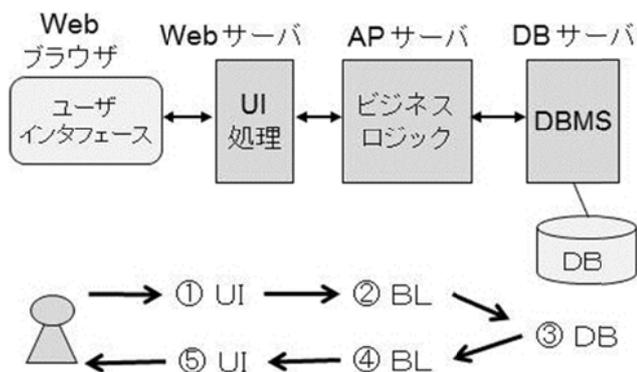


図2. 3層構造とUI駆動型アプローチ

5.2. 提供者と引取者の資格と身元確認

4.2節で取り上げた規則の中でシステムの視点の規

則として以下のものがある。

(A12) もし、申請者が市内在住または市内に勤務しており、かつ18歳以上ならば、登録する

(A2)業者は登録できません

(A32) もし申請者が登録希望の時は身元確認する

これらの規則を統合した場合の主要な処理をテンプレートで定義すると以下ようになる

UI：システムは登録フォームを表示して利用者から要求を受け取る

BL：システムは上記の規則に基づき要求をチェック

DB：システムはDBに登録

BL：システムはDBアクセス結果を受け取る

UI：システムは利用者登録番号を表示

ここでは、エラー処理や身元確認処理や登録番号生成処理の記述は省略している。共通性のあるエラー処理は設計段階で定義されると思われる。本人確認の方法は各自自治体で異なる。すでにインターネットでのアカウントを有するかもしれないし、本システム利用の前に実際に窓口で手続きが必要かもしれない。識別番号の発行方法に関しては、逐次番号生成処理や特定のフォームの定義機能の実装が必要かもしれない。

5.3. 対象物品の登録

4.3節で取り上げた規則の中でシステムの視点の規則として以下のものがある。

(B14)システムは提供者に対し、登録前に、物品が家庭内で使用されていたものであるという誓約を求める
この規則の主要な処理をテンプレートで定義すると次のようになる。

UI：システムは登録フォームを表示して利用者から要求を受け取る

BL：システムは上記の規則に基づき要求をチェック

DB：システムはDBに登録

BL：システムはDBアクセス結果を受け取る

UI：システムは物品登録番号を表示

5.4. 登録できない物品リスト

4.4節で取り上げた規則の中でシステムの視点の規則として以下のものがある。

(C14)システムは提供者に対し、登録前に物品が登録できない物品リストに含まれないという誓約を求める

(C15)登録された物品は、管理者のチェックののちに掲示される

規則(C14)の主要な処理をテンプレートで定義すると以下ようになる。

UI：システムは誓約フォームを表示して利用者から誓約を受け取る

BL：システムは上記の規則に基づき誓約をチェック

DB：(スキップ)

BL：(スキップ)

UI：システムは結果のメッセージを表示

実際にはこの処理は、物品登録フォームの中に登録できない物品リストに関する☑ボックスを加えることで、前節の(B14)の処理に含められる。一方、(C15)の場合は、システム管理者用の次の処理が定義される。

UI：システムは管理者用のフォームを表示してチェック要求を受け取る

BL：システムは仮登録物品のチェックのための DB アクセス要求を作成

DB：システムはDBに検索要求

BL：システムはDB検索結果を受け取る

UI：システムは仮登録物品リストを表示

さらに、以下の処理定義が続く。

UI：システムは管理者から判断内容を受け取る

BL：システムは仮登録物品の本登録または削除のための DB アクセス要求を作成

DB：システムはDBに本登録または削除を要求

BL：システムはDBから結果を受け取る

UI：システムは結果を表示

このような処理定義は、システムの開発者と運用管理者を兼ねるエンドユーザには容易と思われる。

5.5. その他の運用規則について

以下、詳細は省略するが、システムの視点での規則は、4.5 節の登録手順では(D1), (D21), (D31), 4.6 節の登録物品では(E1), (E21), (E31), (E4), (E52), 4.7 節の引取者の決定では(F11), (F21), (F31), (F4), 4.8 節の登録物品の削除では(G11)である。なお、(G11)は物品を自治体が預かる現行方式の規則なので、Web 上での運用の場合は次の規則になる。

(G13)もし、登録物品が5週間経過して残っているならば、登録を削除し、提供者に通知する

いずれも今回導入したテンプレートにより記述可能であった[4]。

5.6. テンプレート利用のケーススタディの考察

インターネット検索で収集した再利用促進サービスのビジネスロジックの一部について、{UI→BL→DB→BL→UI}のテンプレートを適用して要求定義をした。多様なビジネスロジックが存在したが、本テンプレートは要求定義に有用であるという結論を得た。

本技術をドメイン特化型のフレームワークやビジュアルモデリングツールに組み込むことにより、エンドユーザ主導開発が促進されると思われる。

なお、エンドユーザが入力フォームから出力フォームへの変換規則を定義して実行する方式の研究[5]では、抽象フォームという概念を導入して、システム内部の{UI→BL→DB}や{DB→BL→UI}の処理をフォーム変換として定義する方法をとっており、今回のテンプレート方式とは相補的な関係にある。

6. まとめ

エンドユーザ主導開発実現のための主要な技術課題の一つである多種多様なビジネスロジックの定義方法に関して、{UI→BL→DB→BL→UI}テンプレートを用いる方法を提案し、不用品再利用促進サービスのビジネスロジックの定義に用いて、エンドユーザによる要求定義という観点での有用性を確認した。

文 献

- [1] J. C. Brancheau, and C. V. Brown, The management of end-user computing: status and directions, ACM Computing Surveys, vol.25, no.4, pp.437-482, 1993.
- [2] 中所武司, “業務の知識を有するエンドユーザ主導のアプリケーション開発技法 ～フレームワーク・ドメインモデル・サービス連携～,” 電子情報通信学会 技術研究報告, Vol.107, No.331, 知能ソフトウェア工学研究会, KBSE2007-30, 19-24, Nov. 2007.
- [3] 中所武司, “不特定多数向けシステムの要求分析について,” 情報処理学会ソフトウェア工学研究会第35回要求工学ワーキンググループ ワークショップ, Oct. 2010.
- [4] T. Chusho, “Classification and Definitions of Business Logic for End-User-Initiative Development,” The 11th International Conference on Software Methodologies, Tools and Techniques (SoMeT_12), (発表予定), Sep. 2012.
- [5] T. Chusho and N. Yagi, “Modeling by Form Transformation for End-user Initiative Development,” IEEE COMPSAC 2008, pp.331-334, July 2008.
- [6] W. W. Cotterman, and K. Kumar, “User cube: A taxonomy of end users,” Communications of the ACM, vol.32, 11, 1313-1320, 1989.
- [7] G. Fischer, K. Nakakoji and Y. Ye, Metadesign: Guidelines for supporting domain experts in software development, IEEE Software, vol.26, no.5, pp.37-44, Sep./Oct. 2009.
- [8] A. J. Ko, R. Abraham, M. M. Burnett and Brad A. Myers, Guest editors' introduction: End-user software engineering, IEEE Software, vol.26, no.5, pp.16-17, Sep./Oct. 2009.
- [9] S. Minsky, “The Challenge of BPM Adoption,” http://www.ebizq.net/topics/biz_opt/features/5757.html, Mar. 2005.
- [10] 佐藤学, 萩原淳, 斎藤忍, “要件定義書へのビジネスルールの記述方法に関する事例調査と考察,” 電子情報通信学会 技術研究報告, Vol.111, No.396, 知能ソフトウェア工学研究会, KBSE2011-55, 13-18, Jan. 2012.
- [11] J. Sprinkle, M. Mernik, J. Tolvanen and D. Spinellis, Guest editors' introduction: What kinds of nails need a domain-specific hammer?, IEEE Software, vol.26, no.4, pp.15-18, July/Aug. 2009.
- [12] A. Sutcliffe, N. Mehandjiev, (Guest Ed.), End-user development, Communications of the ACM, vol.47, no.9, pp.31-32, 2004.
- [13] K. E. Wiegers, Software Requirements (Second Ed.), Microsoft Press, 2003. (渡部洋子監訳, ソフトウェア要求, 日経BPソフトプレス. 2003)