

存在従属グラフからの機能規模の測定

井田明男[†] 金田重郎[†] 熊谷聡志[†] 矢野寛将[†]
Akio Ida[†] Shigeo Kaneda[†] Satoshi Kumatani[†] Hiromasa Yano[†]

[†] 同志社大学大学院・理工学研究科

[†] Graduate School of Science and Engineering, Doshisha University

要旨

ソフトウェアの機能規模の測定方法として国際規格の COSMIC 法がある。この方法は、すべての機能プロセスにおけるデータの移動を計測しなければならないため、利用者機能要求が機能プロセスを取り出せるほど詳細でない場合には適用が難しい。それに対して、業務アプリケーションの要求記述は、機能に関する記述の網羅性は概して高くない。そこで、本稿では、COSMIC 法をベースに、業務で扱うエンティティの存在従属性に着目した機能規模の測定法を提案する。要求記述から先にエンティティの存在従属グラフを作成し、そこから機能プロセスを抽出して測定を実施する。そのため、利用者機能要件の取りこぼしが少なく、正確な機能規模の測定が行えると期待できる。例題の要求記述について、提案手法による測定結果と COSMIC 法による測定結果を比較した結果、それらの間には高い一貫性が得られたため、提案手法は有効であると判断する。

1. はじめに

業務アプリケーション^a開発の現場において、開発対象の機能規模を簡単かつ正確に測定することへの期待が高まっている。さらに、近年では、スコーブを細分化して、小さなリリースを繰り返すスタイルが広まってきたため、見積りの頻度も高くなる傾向にある。ソフトウェアの規模をどのように測るかについては予めから種々の議論が展開されている領域であり、いくつもの測定手法が提案されている。ソースコードの行数で規模を測定するのは確実な方法であるが、実装が終わってからでないと測定できないため、見積りの用途には使えない。

一方、あらかじめモデルを作成してソフトウェアの規模を測る方法としては、ファンクション・ポイント (IFPUG : International Function Point Users Group) 法[1], COSMIC (COmmon Software Measurement International Consortium) 法[2]などが知られている。中でも COSMIC 法はユースケース・モデルと親和性が高く、ISO で承認され JIS 標準にもなっている測定手法であるため、筆者らは機能規模を測定する必要がある時には COSMIC 法を用いている。

COSMIC 法は認知された測定手法であるが、その精度を保つためには、イベントフローレベルのユースケースの論理的な実現の分析を必要とする。しかしながら、実用的な観点からはもっと早期に、すなわち、ユースケース候補の中から開発対象をユースケースとして決定する時点で機能規模に関する情報が得られることが望まれる。

そこで、本稿では、COSMIC 法をベースに、業務アプリケーション専用としてカスタマイズした測定手法を提案する。カスタマイズには、その業務ドメインで扱うエンティティの存在従属性に着目した。なぜならば、存在従属性はインスタンスのライフサイクルに基づく関係であるため、それらを扱う機能の時間的前後関係も半ば規定するからである。そのことを自然な形で取り入れた提案手法は、要求定義の早い段階で適用でき、測定も簡単かつ正確であり、その作業成果物も機能的規模の測定の使用だけでなく、機能要件の網羅性確認にも優れたものである。

以下、第 2 章では、COSMIC 法の概要と課題を説明する。第 3 章では提案手法について説明する。第 4 章では、検証実験として COSMIC 法と提案手法の結果を比較する。第 5 章はまとめである。

2. COSMIC 法

2.1. COSMIC 法の概要

COSMIC 法はソフトウェアの機能規模を測定する手法である。1997 年に Full Function Point (FFP) 法 (ver1.0) という名前でリアルタイム、技術、システムソフトウェアの機能規模を測定する手法として提案された。その後 2002 年に ISO で承認され、2006 年には JIS X0143 として JIS 標準となった手法である[2]。

2.2. COSMIC 法の測定要素

COSMIC 法では、図 1 に示すように測定対象ソフトウェアについて、システム境界をまたがったデータ移動数と永続化

^a 取引等に関する事実を効率的に捕捉、蓄積、管理するためのアプリケーションと緩やかに定義する。その特性上、データベース管理システムを核としたアーキテクチャである。

記憶域に読み書きするデータ移動数の合計値を求めることで利用者の立場から見た機能の規模である機能規模を測定する。測定するデータ移動の種類は表1に示す4種類である。移動するのは、データ・グループと呼ばれるデータのまとまりで、論理データモデルのエンティティに相当する。データ移動の対象となるデータ・グループの種類数を機能プロセスと呼ばれる単位毎に数えていくことによって、最終的にソフトウェア全体の機能規模を求めることができる[2]。

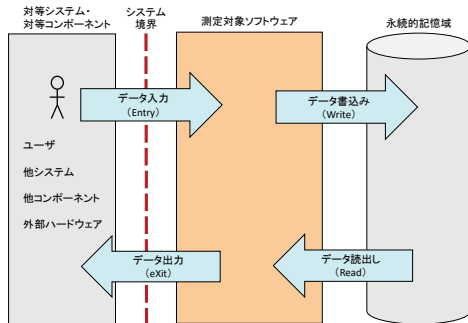


図1: COSMIC法の要素間のデータ移動モデル

表1: COSMIC法におけるデータ移動の種類

データ移動の種類	説明
エントリ (Entry)	システム境界外からユーザインタフェース (UI) や通信を通じて入力されることに対応するデータ移動
エクジット (eXit)	UI や通信によりシステム境界外に出力されることに対応するデータ移動
リード (Read)	永続化記憶域の注目オブジェクト (エンティティ) から読み出されるデータ移動
ライト (Write)	永続化記憶域の注目オブジェクトに対して書き込まれるデータ移動

2.3. COSMIC法による測定手順の概略

COSMIC法による測定手順は文献[2]を参照されたい。システムの境界と利用者機能要求を定義し、次に、利用者機能要求を構成する機能プロセスを明らかにする。そして、利用者機能要求毎の機能規模とソフトウェアの機能規模を求める。

2.4. COSMIC法適用の課題

筆者らは、COSMIC法を適用するための課題は2つあると考えている。

要求定義プロセス上の測定時期に関する課題

1つ目の課題は、機能規模を測定可能な要求定義プロセス上の時点に関するものである。COSMIC法で正確な測定を行うためには、要求定義の後半まで待たねばならない。前述のとおりCOSMIC法では、アクターの入力からシステムの応答までを1つの機能プロセスとし、その中で行われるデータ移動を計測するためである。

要求記述の特性に関する課題

2つ目の課題は、要求記述の特性に関するものである。大抵の要求記述は、何を管理するかについてはある程度言及されているものの、どのように管理を実現するかについては捨象されていると考えられる。そのため、これらの課題を克服すべく、筆者らは要求定義の早い時点でも正確な機能規模が測定できるように、COSMIC法のカスタマイズを検討することにした。

3. 提案手法

3.1. 提案手法の概要

COSMIC法が要求する入力モデルは、ユースケース記述としてのイベントフローレベル以上の詳細さを有する必要がある。そこで、提案手法では、ユースケース・モデルではなく、業務についての要求記述から存在従属グラフを先に作成し、それを測定のための入力とする。留意すべきは、要求記述は、あくまでも作成者が業務の一部分だけをプロファイルして描写した結果であるため、ストレート・フォワードに機能規模測定のための機能プロセスに展開することはできないことである。

3.2. 提案手法の手順

Step 1: エンティティの存在従属グラフを作成する

エンティティとは、業務で管理すべき重要な概念である。存在従属性の概念は P. チェンが用いている[3]。エンティティの存在従属グラフは、エンティティ間の存在従属性に着目して、作成する有向グラフである。UMLクラス図の記法を流用している。日本語要求記述文からのエンティティの識別については文献[4]を、存在従属という概念、および存在従属グラフ作成の詳細については、文献[5]を、それぞれ参照されたい。図2は典型的な受注と出荷のドメインについて作成したエンティティの存在従属グラフである。図2は受注は出荷に先立つといった業務上の制約を表現している。なお、以降の説明では、COSMIC法でいうデータ・グループは、提案手法では、エンティティの全属性またはエンティティの一部の属性を摘み集めたものに相当し、COSMIC法でいう注目オブジェクトは、永続化記憶域に格納される正規化されたデータであると定義されている[2]ため、提案手法ではエンティティそのものに相当すると見做す。

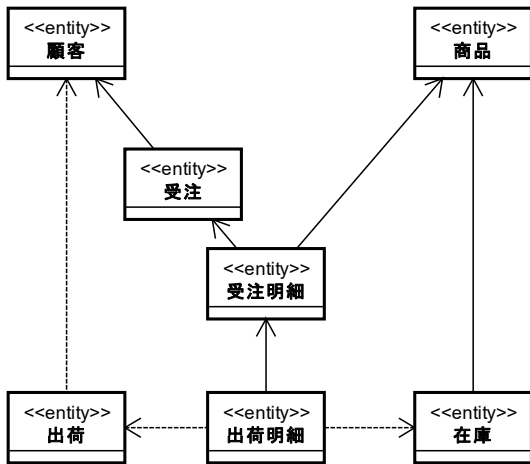


図 2：エンティティの存在従属グラフ

表 3：ドメイン管理要件を利用者機能要件に展開した表

ドメイン管理要件	FUR#	利用者機能要件(ユースケース)
顧客を管理する	1.1	顧客を新規登録する
	1.2	顧客を検索・同定する
	1.3	顧客の属性を更新する
	1.4	顧客を抹消する
商品を管理する	2.1	商品を新規登録する
	2.2	商品を検索・同定する
	2.3	商品の属性を更新する
	2.4	商品を抹消する
受注を管理する	3.1	受注を新規登録する
	3.2	受注を検索・同定する
	3.3	受注の属性を更新する
	3.4	受注を抹消する
出荷を管理する	4.1	出荷を新規登録する
	4.2	出荷を検索・同定する
	4.3	出荷の属性を更新する
	4.4	出荷を抹消する
在庫を管理する	5.1	在庫を新規登録する
	5.2	在庫を検索・同定する
	5.3	在庫の属性を更新する
	5.4	在庫を抹消する

Step 2：存在従属グラフから利用者機能要件を求める

エンティティの存在従属グラフが得られたならば、それをもとに COSMIC 法でいう利用者機能要件を導出する。1 種類のエンティティの管理には、CRUD に対応する 4 つの機能要件が必要である。エンティティが見出しと明細に分れている場合は、それらは概念としては 1 つであると見做すと、例題の場合は、5 種類のエンティティが識別されるため、自動的に 20 の利用者機能要件が導かれる。表 3 はこれらを一覧した表である。なお、筆者らは 1 種類のエンティティに対する管理要件をドメイン管理要件と呼んでいる。

Step 3：利用者機能要件を機能プロセスに展開する

COSMIC 法でいう機能プロセスは、必ず複数種類のデータ移動が続けて実行されるものであるとされる[2]。ここでいうデータ移動とは、2.2 節で言及した COSMIC 法の測定対象となる表 1 の 4 種類である。

機能プロセスも存在従属グラフからある程度は導くことが可能である。筆者らは、存在従属グラフからユースケースおよびユースケース記述を逆生成する提案を行っている[6]。この提案によれば、存在従属性は、エンティティのインスタンス間に生成制約と削除制約を与える。これらの制約から、例題の場合、図 2 が示すとおり、受注は、顧客に存在従属し、受注明細は商品に存在従属するため、受注を新規登録するには、表 4 の機能プロセスが必要であることが導かれる。受注を捕捉するには、あらかじめ存在する顧客のインスタンスと商品のインスタンス群を新規の受注のインスタンスを生成して結び付けなければならない。

表 4：利用者機能要件を機能プロセスに展開した表

FUR#	利用者機能要件(ユースケース)	FP#	機能プロセス
2.1	受注を新規登録する	2.1.1	顧客を同定する
		2.1.2	受注を新規登録する
		2.1.3	商品と数量を同定する
		2.1.4	受注明細を新規登録する

存在従属グラフは、エンティティのインスタンスのライフサイクルに関する制約を表現しているため、存在従属グラフが得られれば、ドメイン管理要件、利用者機能要件、および、利用者機能要件を実現するための機能プロセスも自動的に導かれることは注目に値する。しかしながら、提案手法では、機能プロセスまで考慮する必要はない。その代わりに、機能プロセスを考慮して、あらかじめ利用者機能要件タイプ毎に CFP 値を計算した値を適用することができる。

Step 4：存在従属グラフから利用者要件毎の CFP 値を求める

存在従属グラフと利用者機能要件の一覧が得られれば、利用者機能要件毎の CFP 値は、表 5 によって求めることができる。表 5 は、利用者機能要件のタイプ別に存在従属グラフに登場するエンティティの制約を考慮して筆者らがあらかじめ CFP 値を計算したものである。このように、あらかじめ CFP 値が計算できるのは、存在従属グラフの制約があるからである。表 5 の使い方は以下のとおりである。

1. Step 3 で明らかにした利用者機能要件に着目する
2. 利用者機能要件が扱うエンティティの特性を Step 1 で作成した存在従属グラフから確認する
3. 表 5 を参照しながら、着目している利用者機能要件のタイプに応じた CFP を求める

たとえば、例題の表 4 について実施した場合、「受注を新規登録する」利用者機能要件は、表 5 の FP タイプ#6 に該当し、

受注の n (存在従属している親エンティティの数) は 1 であるため、その CFP 値は 6 である。この作業を、表 3 のすべての利用者機能要件について適用すると表 6 を得る。

表 5: 存在従属性から利用者機能要件タイプ毎にあらかじめ計算で求めた CFP 値の表

FURタイプ#	利用者機能要件(FUR)タイプ	Entry	Read	Write	eXit	CFP値
1	あるエンティティのインスタンスを検索・同定する	同定するためのIDまたはキーワードを入力する	そのエンティティの該当するインスタンスを検索する	なし	同定に成功したインスタンスの諸属性を表示する 同定に失敗したメッセージを表示する	4
2	親を持たない独立クラスのあるエンティティのインスタンスを新規登録する	新規登録するインスタンスの諸属性値を入力する	重複登録を避けるためエンティティのインスタンスを検索する	新規インスタンスを書き込む	登録に成功したインスタンスの諸属性を表示する 登録に失敗したメッセージを表示する	5
3	親を持たない独立クラスのあるエンティティのインスタンスの属性値を更新する	更新対象のインスタンスを同定するために機能プロセスタイプ#1を実行する。そのためのCFP値は4 属性値を更新するインスタンスの諸属性値を入力する	なし	属性値変更後のインスタンスを書き込む	更新に成功したインスタンスの諸属性を表示する 更新に失敗したメッセージを表示する	8
4	あるエンティティのインスタンスを削除する。ただし、そのエンティティに存在従属したり、そのエンティティを参照するエンティティはグラフ上に存在しない	削除対象のインスタンスを同定するために機能プロセスタイプ#1を実行する。そのためのCFP値は4 削除の実行を入力する	なし	同定されたインスタンスを削除する	削除に成功したインスタンスの諸属性を表示する 削除に失敗したメッセージを表示する	8
5	あるエンティティのインスタンスを削除する。ただし、そのエンティティに存在従属したり、そのエンティティを参照するn種類のエンティティがグラフ上に存在する	削除対象のインスタンスを同定するために機能プロセスタイプ#1を実行する。そのためのCFP値は4 削除対象のインスタンスに存在従属、または削除対象のインスタンスを参照しているインスタンスを同定するために、削除対象のエンティティに存在従属、または削除対象のエンティティを参照しているエンティティの種類数nだけReadする。そのためのCFP値はn 削除の実行を入力する	なし	同定されたインスタンスを削除する	削除に成功したインスタンスの諸属性を表示する 削除に失敗したメッセージを表示する	n+8
6	n種類の親を持つ、ある存在従属エンティティのインスタンスを新規登録する	親エンティティのインスタンスを同定するために、種類数nだけエンティティをReadする。そのためのCFP値はn 自分自身を新規登録するために機能プロセスタイプ#2を実行する。そのためのCFP値は5				n+5
7	n種類の親を持つ、ある存在従属エンティティのインスタンスの属性値を更新する	親エンティティのインスタンスを同定するために、種類数nだけエンティティをReadする。そのためのCFP値はn 自分自身の属性値を更新するために機能プロセスタイプ#3を実行する。そのためのCFP値は8				n+8

表 6: 利用者機能要件毎の CFP 値

ドメイン管理要件	管理対象の親クラスの数	FUR#	利用者機能要件(≒ユースケース)	CFP値
顧客を管理する	0	1.1	顧客を新規登録する	5
		1.2	顧客を検索・同定する	4
		1.3	顧客の属性を更新する	8
		1.4	顧客を抹消する	10
商品を管理する	0	2.1	商品を新規登録する	5
		2.2	商品を検索・同定する	4
		2.3	商品の属性を更新する	8
		2.4	商品を抹消する	10
受注を管理する	2	3.1	受注を新規登録する	7
		3.2	受注を検索・同定する	4
		3.3	受注の属性を更新する	10
		3.4	受注を抹消する	10
出荷を管理する	1	4.1	出荷を新規登録する	6
		4.2	出荷を検索・同定する	4
		4.3	出荷の属性を更新する	9
		4.4	出荷を抹消する	8
在庫を管理する	1	5.1	在庫を新規登録する	6
		5.2	在庫を検索・同定する	4
		5.3	在庫の属性を更新する	9
		5.4	在庫を抹消する	9
合計				140

このシステムは、インターネットから日本国内のビジネスホテルの宿泊プランの検索と宿泊プランの予約を行うことができる。宿泊プランの検索は会員でなくても可能であるが、予約は会員でないとできない。利用者が宿泊を希望する地区(都道府県, エリア), 利用期間(チェックイン日付, および泊数)を指定すると、指定した地区に所在する提携ビジネスホテルの一覧が表示される。利用者が一覧に表示されたホテルを選択すると、指定されたホテルの指定された宿泊期間に予約可能な宿泊プランの一覧が表示される。宿泊プランとは、部屋グレード、部屋ベッドタイプ、喫煙可否、食事オプションの組み合わせに名前を付けたものである。利用者が一覧から希望する宿泊プランを選択すると、会員の認証が求められる(認証済みでない場合)。会員はあらかじめ登録した会員IDとパスワードを入力する(この際、会員登録が済んでいない場合は、新規登録を行うこともできる。その場合、利用者は氏名、住所、メールアドレス、パスワードを登録する)。認証が成功すると、選択された宿泊プランの詳細、チェックイン日、泊数、チェックイン日からチェックアウト日にかけての日毎の宿泊料金、そして宿泊料金の合計金額が表示される。会員が表示内容を確認し、予約を実行すると、宿泊プランが引当てられ(ただし、具体的な部屋の割り当てまでは行われない)、予約番号が表示されるとともに、予約番号と予約内容が記載された電子メールが会員の登録済みアドレス宛に届く。

図 3: サンプルプロジェクトの要求記述

表 6 からは、図 2 の存在従属グラフを扱うのに 140 の CFP 値に相当する規模が必要になることも分る。

4. 検証実験

この章では、ある程度の大きさと複雑さを有する業務アプリケーションに対して提案手法を適用して機能規模を測定した結果と、COSMIC 法を本来の手順で適用して機能規模を測定した結果を比較する。測定対象には、ビジネスホテル宿泊予約サイトの事例を用いた。インターネットから日本国内の提携ホテルの宿泊プランの検索と予約を行うことができる。図 3 に宿泊予約サイトの詳細な要求記述を示す。

4.1. 提案手法による測定

本節では、図 3 の要求記述に対して、3.2 節で述べた提案手法の手順を適用する。具体的には、1) 存在従属グラフの作成、2) ドメイン管理要件の識別、3) 利用者機能要件の識別、4) 表引きによる利用者管理要件毎の CFP 値の取得の順で測定を進める。図 4 に要求記述から作成したエンティティの存在従属グラフを示す。表 7 に存在従属グラフから導出したドメイン管理要件、利用者機能要件および、表 5 を参照することによって求めた機能プロセス毎の CFP 値を示す。

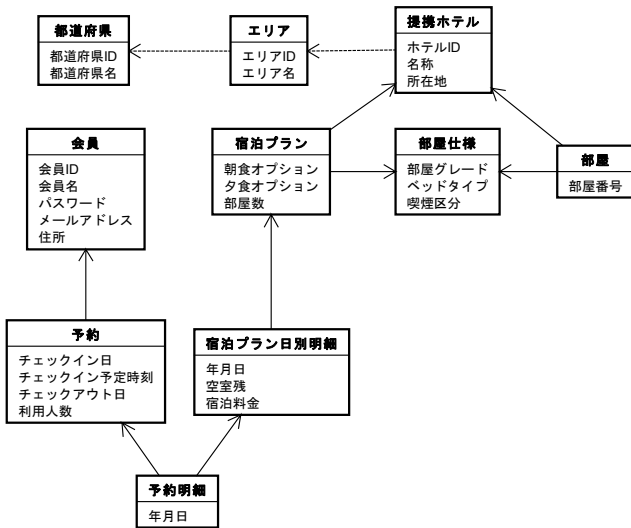


図 4：エンティティの存在従属グラフ

表 7：提案手法で求めた CFP 値

ドメイン管理要件	管理対象の親クラスの数n	利用者機能要件	提案手法で求めた CFP値
会員を管理する	0	会員を新規登録する	5
		会員を検索・同定する	4
		会員を更新する	8
		会員を抹消する	9
予約を管理する	2	予約を新規登録する	7
		予約を検索・同定する	4
		予約を更新する	8
		予約を抹消する	8
宿泊プラン日別明細を管理する	1	宿泊プラン日別明細を新規登録する	6
		宿泊プラン日別明細を検索・同定する	4
		宿泊プラン日別明細を更新する	9
		宿泊プラン日別明細を抹消する	9
宿泊プランを管理する	2	宿泊プランを新規登録する	7
		宿泊プランを検索・同定する	4
		宿泊プランを更新する	8
		宿泊プランを抹消する	9
部屋仕様を管理する	0	部屋仕様を新規登録する	5
		部屋仕様を検索・同定する	4
		部屋仕様を更新する	8
		部屋仕様を抹消する	9
部屋を管理する	2	部屋を新規登録する	7
		部屋を検索・同定する	4
		部屋を更新する	8
		部屋を抹消する	8
提携ホテルを管理する	1	提携ホテルを新規登録する	6
		提携ホテルを検索・同定する	4
		提携ホテルを更新する	9
		提携ホテルを抹消する	10
エリアを管理する	1	エリアを新規登録する	6
		エリアを検索・同定する	4
		エリアを更新する	9
		エリアを抹消する	9
都道府県を管理する	0	都道府県を新規登録する	5
		都道府県を検索・同定する	4
		都道府県を更新する	8
		都道府県を抹消する	9
		要求記述に直接関係するCFP値合計	53
		CFP値合計	245

表 7 の CFP 値は、利用者機能要件から、直ちに表 5 と存在従属グラフを参照して求めたものである。提案手法では、このサイトのソフトウェアの機能規模は 53 と測定された。

なお、機能規模の測定とは直接的には関係ないが、要求記述からエンティティの存在従属グラフをまず最初に作成することは、ドメイン管理要件を明確にし、それらの CRUD の必要性から利用者機能要件がほぼ求まるため、要求定義プロセスにおける機能要求の確認のために非常に有効な方法であると思われる。

4.2. COSMIC 法による測定

今度は、図 3 の要求記述に対して、2.2 節で説明した COSMIC 法の手順を適用する。COSMIC 法では、最初に利用者機能要件を明確にする必要がある。利用者機能要件は、UML のユースケースに相当するとされる[2]ため、最初にユースケース図、ユースケース毎にイベントフローを作成し、ユースケースを論理的に実現するための機能プロセスを洗い出す。これは、結構労力の必要な作業である。もちろん、開発が決定したユースケースに対してこの作業を行うのは必要であるが、作り込みを行うかどうか定まっていないユースケース候補に対してこの作業を行うのは避けたいところである。しかしながら、COSMIC 法で機能規模の測定を行うのであればそうはいかない。

表 8 は、おのおのの機能プロセスと機能プロセスが移動するデータ・グループの種類を数えたものである。利用者機能要件欄には、ユースケース分析で明らかにしたユースケース名を記入した。機能プロセス欄には、ユースケース毎に作成したイベントフローで明らかになった、アクターの入力からそれに対するシステムの応答までの対話の一往復を記入してある。そして、エントリ欄からエクジット欄間のデータ・グループは図 4 の存在従属グラフに記載されているエンティティをそのまま使用した。エクジット欄で X(eXit)が複数マークされている機能プロセスは、アクターへの通常の応答以外にもエラーメッセージの出力とメール送信があることを示している。そして、CFP 値欄は、その行に含まれる E, R, W, X の個数を単純にカウントした値である。

ここで、表 7 と表 8 では、異なる手法で CFP 値の合計を求めているにも関わらず、提案手法が 53 で、COSMIC 法は 51 と極めて近い値になっている。また、表 9 は、表 7 と表 8 の値を、ドメイン管理要件毎に集計して比較したものであるが、こちらも極めて近い値になっている。ただし、COSMIC 法に用いた測定では、機能プロセスに近視眼的にならないように

注意しなければならない。なぜならば、表7中でグレイアウトしてある利用者機能要件は、要求記述からは直接読み取りにくいものの必ず必要な要件のはずである。提案手法では、ひとまず存在従属グラフに変換し、利用者機能要件はそこから導出するため、このような利用者管理要件も取りこぼさないが、COSMIC法では取りこぼしてしまう可能性がある。これは、高速道路の建設に例えれば、本線の見積りに気を奪われ、パーキングやインターチェンジの存在を見落とすようなものである。

表8：COSMIC法で求めたCFP値

利用者機能要件	機能プロセス	トリガ・エントリ	会員	都道府県	地区	提携ホテル	宿泊プラン	部屋仕様	宿泊プランの日別明細	予約	予約明細	エクジット	CFP値
会員を登録する	アクターがメニューから会員登録を選ぶと、会員登録フォームが表示される	E										X	2
	アクターが会員属性を入力すると、当該会員が新規登録され、登録完了メッセージが表示される	E	W									X X X	5
会員を認証する	アクターが会員IDとパスワードを入力すると、認証の成否が表示される	E	R									X X	4
ホテルを検索する	アクターが都道府県とエリアを指定すると、当該エリアに存在する提携ホテルの一覧が表示される	E		R	R	R						X X	6
宿泊プランを検索する	アクターが、宿泊を希望する地区(都道府県, エリア), 利用期間(チェックイン日付, および泊数)を指定すると、システムは「ホテルを検索する」を実行する	E	ホテルを検索するでカウント済みの6										7
	アクターがホテルを選択すると、指定されたホテルの指定された宿泊期間に予約可能な宿泊プランの一覧が表示される	E				R	R	R	R			X X	7
宿泊プランを予約する	アクターが宿泊プランを選択すると、システムは「会員を認証する」を実行する	E	会員を認証するでカウント済みの4										5
	認証が成功すると、選択された宿泊プランの詳細、チェックイン日、泊数、チェックイン日からチェックアウト日にかけての毎日の宿泊料金、そして宿泊料金の合計金額が表示される	E				R	R	R	R			X X	7
	アクターが予約を実行すると、宿泊プランが引当てられ(ただし、具体的な部屋の割り当てまでは行われない)、予約番号が表示されるとともに、予約番号と予約内容が記載された電子メールが会員の登録済みアドレス宛に届く。	E								R W	W	W	X X X
												CFP値合計	51

表9：COSMIC法で求めたCFP値の比較

ドメイン管理要件	提案手法で求めたCFP値	COSMIC法で求めたCFP値
会員を管理する	9	11
予約を管理する	44	40

5. まとめ

本稿では、業務アプリケーションの機能規模測定のためにエンティティの存在従属性に着目したグラフを導入することによってCOSMIC法をカスタマイズする手法を提案した。COSMIC法を開発対象やプロジェクトの事情に合うようにカスタマイズする提案はすでにいくつか存在しているが、エンティティの存在従属性に由来する制約を、機能プロセスへの展開ルールとして位置づけ、扱うエンティティの存在従属グラフ上の位置に着目することによって機能規模の測定に結び付けたところに本提案手法のオリジナリティがある。ただし、提案手法には、特に表5などはまだまだ改良の余地が残されていると思われる。それらは現場で提案手法を実際に運用する中で、改良していきたい。また、運用の中でCFP値と実際のコストの関係も明らかにしていきたい。

参考文献

- [1] Albrecht A.J., Gaffney Jr. J.E. : "Software function, source lines of code and development effort prediction: a software science validation, IEEE Transactions on Software Engineering, Vol. SE-9, pp. 639-648 (1983/11)
- [2] The Common Software Measurement International Consortium : COSMIC 機能規模測定法 Version 3.0 手法概要編・測定マニュアル, 日本ファンクションポイントユーザ会 (2007/9)
- [3] P. Chen : The Entity Relationship Approach to logical Database Design, QED information science, Wellesley (1979/1)
- [4] 金田重郎, 井田明男, 酒井孝真, 熊谷聡志 : 日本語仕様文からの概念モデリングガイドライン—行為文と関数従属性に基づくクラス図の作成, 電子情報通信学会論文誌 D, Vol.J98-D, No.7, pp.1068-1082 (2015/7)
- [5] 井田明男, 金田重郎, 熊谷聡志, 藤本明莉 : 存在従属性に着目した論理要件ロバストなドメインモデルの作成—ドメインクラス図をユビキタス言語として用いるために, 情報処理学会論文誌, 56巻5号 pp 1340-1350 (2015/5)
- [6] 金田重郎, 井田明男, 矢野寛将 : 存在従属に基づくユースケースの逆生成, 電子情報通信学会, 知能ソフトウェア研究会, 信学技報, vol. 115, no. 54, KBSE2015-3, pp. 13-18 (2015/5)