

PBL 型システム開発におけるサービス指向設計の試みと評価

梶原 雄一郎†¹ Rathnamalala Lasith†¹ 大場 みち子†²

†¹筑波大学 大学院 システム情報工学研究科 コンピュータサイエンス専攻
〒305-8577 茨城県つくば市天王台 1-1-1

†²株式会社 日立製作所 ソフトウェア事業部 〒244-8555 横浜市戸塚区戸塚町 5030 番地

E-mail: †¹kaji@kslab.cs.tsukuba.ac.jp, lasith@netlab.cs.tsukuba.ac.jp,

†²michiko.oba.cq@hitachi.com

概要

PBL(Project Based Learning)型システム開発授業の演習において大学内の書籍部用の書籍注文システムの開発・構築を行った。その際、ソフトウェア設計の段階でサービス指向に基づかない独自手法によって設計を行ったが、システム実装後に発展課題としてシステムの拡張機能をサービス指向に基づいて設計を行った。本論文では、独自手法を用いた設計とサービス指向に基づく設計を行った場合の違いを比較・評価し、考察を行う。

Evaluation of book ordering system designed by service oriented architecture in PBL type system development

Yuichiro KAJIWARA†¹ Lasith RATHNAMALALA†¹ Michiko OBA†²

†¹Computer Science, Graduate School of System and Information Engineering, Tsukuba
University 1-1-1 Tennodai, Tsukuba, Ibaraki, Japan 305-8577

†²Software Division, Hitachi Ltd. 5030 Totsuka-chou, Totsuka-ku, Yokohama, Kanagawa, 244-8555
Japan

Abstract

The authors had constructed a book ordering system for university book shops as a case study problem in the PBL (Project Based Learning) class. The system had been designed and implemented by our original system design method, which is not based on the service oriented architecture. This paper designs and evaluates the enhanced feature of the book ordering system based on the service oriented architecture and compares the original system design with the service oriented one.

1. はじめに

筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻において「先導的 IT スペシャリスト育成推進プログラム」に採択された「高度 IT 人材育成のための実践的ソフトウェア開発専修プログラム」が開設されている[1]. このプログラムは、IT 高度利用・活用のための専門的スキルと、企業経営に関する知識を備えたプロジェクトマネージャ、IT アーキテクト、IT コーディネータを育成することを目的としたものであり、実践的なスキルを身につけるべく、実習型の特殊な学習カリキュラムが組まれている。

その一環として、PBL 型のシステム開発の演習があり、実際に可動することを目的とした大学書籍部用のシステム開発を行った[2]. この PBL 型のシステム開発の演習においては、システム提案から、要求定義、外部設計、内部設計、実装、試験までの各工程を 4 人のチームで行った。

一方、近年において、ソフトウェアの機能を一つのサービスとし、ネットワーク上でそれら複数のサービスを組み合わせてシステムを構築する SOA(Service Oriented Architecture:サービス指向アーキテクチャ)の考え方に注目が集まっている[3]. PBL 型システム開発においては独自手法によって設計を行ったが、開発したシステムの拡張機能としてサービス指向の設計を試みた。本論文では、それらの試みとサービス指向設計による有用性等の評価について考察を述べる。

2. 従来型システム開発と SOA

本章では、従来型のシステム開発に存在するいくつかの問題点と、それらの問題に対して近年注目を集めている SOA の考え方および、その利点や特性について述べる。

2.1. 従来型のシステム開発

近年、ビジネスを取り巻く環境が急激に変化している。ユーザ企業としては、変化に合わせて素早く柔軟に新たなビジネスモデルを実現するために、早急にシステムの改善を行いたいというニーズがある。しかし、従来型のシステムは、規定された業務を処理することに固定され、システムの変更を柔軟に行うことが難しい。従って、ビジネスの変化に応じて新しいビジネスロジックを追加したり、他のシステムと連携させたり、処理フローを変更することは、非常に困難であり、コストと時間がかかる。システムがビジネスを支える基盤でありながら変化に柔軟に対応できないことは、従来型システムの大きな問題点である。また、既存のシステムの変更が困難な場合、新たにシステムを構築することが考えられるが、その場合、複数の部門で同じ機能を実現するアプリケーションが重複して開発・運用されるようになり、重複投資となりやすい側面がある。このような問題を解消しながら、新たなビジネスモデルを素早く柔軟に構築するためには、変化に対応したアーキテクチャが必要となる。そのような中で注目されているのが、SOA である。以下 SOA の特徴について述べる。

2.2. SOA

SOA とは、複数のサービスを結合し、システム全体を構築する手法および概念である。サービスとは、ビジネス的に意味のある単位で切り出したシステム機能のことを表す。

以下では、2.1 節で述べた従来型システム開発における問題が SOA を用いるとどのように解決されるかについて、SOA の 4 つの特性を用いて説明する。

柔軟性：サービスを組み合わせることでシステムを構築する場合に、サービスとその順番を自由に選択することができる。また、ビジネス環境の変化に応じてシステムを変更することも容易である。

拡張性：必要に応じてサービスを追加することが容易であり、新たなビジネスを開始することが可能である。

再利用性：一つのサービスを様々なシステムから呼び出すことができ、システムごとに類似したアプリケーションを開発する必要がなくなり、重複投資による無駄がなくなる。

可視性：システムの機能をビジネス単位で切り出し、整理された形でシステムを表現することができ、可視性を高めることができる。

以上、SOA の主な特徴を述べたが、その他にもサービスの標準化やサービスの共有による最適なシステムが構築できる、開発期間や開発コストが削減できる、運用や管理コストが削減できるといった特徴もある。

3. 独自手法による PBL 型システム開発

本章では、PBL 型システム開発の授業において開発を行ったシステムの概要とその設計について述べる。これらの開発は従来型のシステム開発に基づくものである。

3.1. 対象システムの概要

PBL 型システム開発においては、筑波大学内の書籍部を対象としてシステム開発を行った。実際に書籍部や書籍部の運営会社へヒアリングに伺い、どの領域に関して開発を行うかを検討した結果、私たちは大学書籍部が行っている業務の中で特に書籍注文の業務に関して注目することにした。

筑波大学書籍部において、書籍の注文を行うためには、直接売店へ行って注文用紙に記入する必要がある。その際、顧客は注文用紙に記入する際、書籍名や出版社などの情報をあらかじめメモ等しておく必要があり、手間のかかる作業となっていた。また、書籍到着の連絡方法は、電話で行っており、顧客が電話に出られない事や、発信者不明のため電話に出ないなどの場合があった。

それらの問題に対し、私たちは書籍注文システムの提案を行うと共に開発を行った。

書籍注文システムは、これまで注文内容を売店に出向いて注文用紙に手書きしていたものを Web サイト上から行えるようにしたものである。書籍を注文したい顧客(筑波大学の学生および職員)は書籍部の Web サイト上で注文する書籍のタイトルや出版社などの情報を入力することで書籍の注文を行うことができ

る。また、それらの注文情報はデータベースに格納され、注文管理システムによって管理を行うことができ、書籍が到着した際の連絡もメール配信によって行うことが可能となる。

3.2. システム設計について

開発プロセスとしては、一般的なウォーターフォールモデルに基づき、要求定義、外部設計、内部設計、実装、試験の順に開発を行った[4][5]。ドキュメントとしては、要求定義工程においては、ユースケース図やユースケース記述、外部設計工程では画面遷移図やインタフェース定義書、内部設計工程ではクラス図等を作成した[6]。はじめてのシステム開発であったこともあり、自分たち独自の設計方針で設計を行った。

これらのシステム開発は従来型のシステム開発方式であり、SOA の考えに基づくものではなかった。図 3.1 は、実際に設計の際に作成したユースケース図である。図を見るとわかるようにひとつのシステムにすべての機能がまとめられており、複数のユースケースが混在している。各機能がうまく整理されておらず、わかりにくいものとなっている。

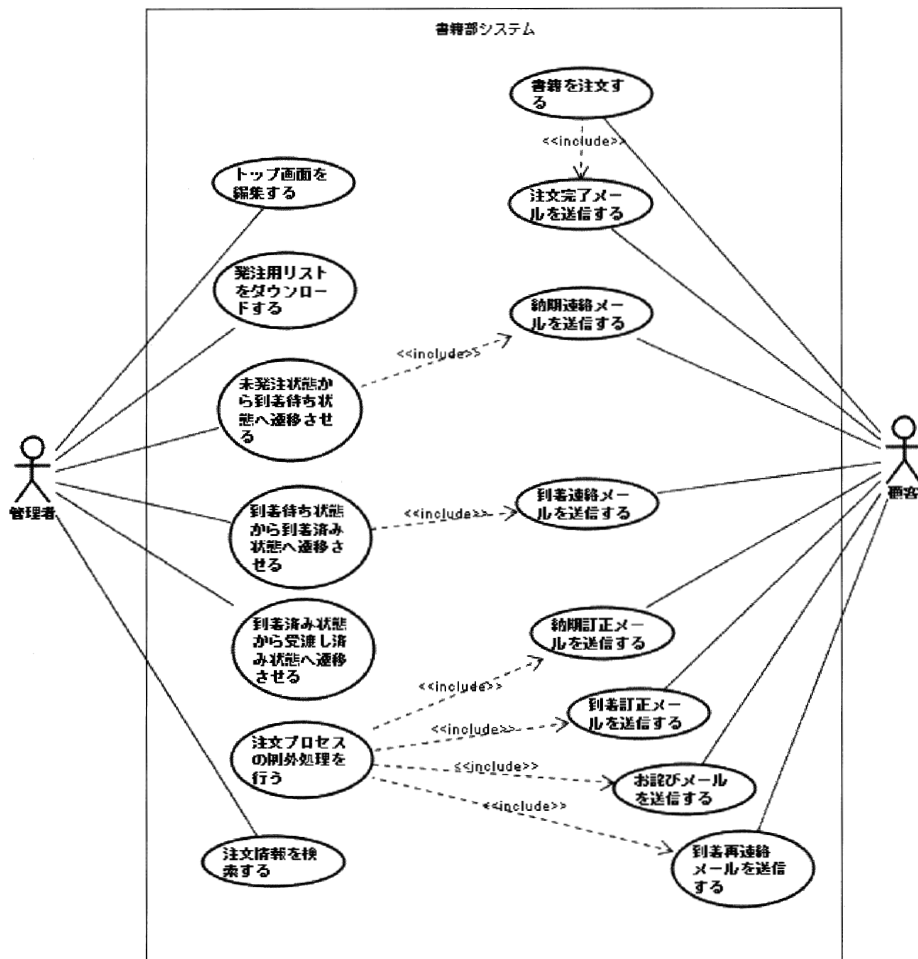


図 3.1 ユースケース図(独自手法)

図 3.2 は複数あるシステム機能の中で、書籍を注文する機能に対して作成したクラス図である。特徴と

しては、注文に必要な顧客情報や書籍情報などのデータを保持するためのクラスと処理を行うクラスがすべて同じパッケージとしてまとめられており、全体としてパッケージの粒度が大きいと言える。

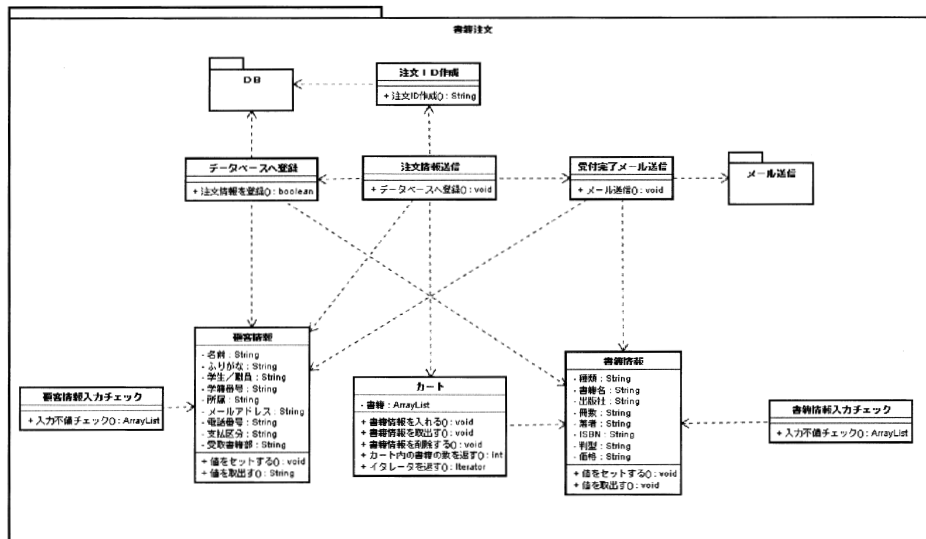


図 3.2 クラス図(独自手法)

4. サービス指向設計の試み

本章では、サービス指向設計を試みた背景と行った設計の詳細について述べる。

4.1. サービス指向設計の試みの背景

「高度 IT 人材育成のための実践的ソフトウェア開発専修プログラム」のカリキュラムでは、実践的なスキルを身に付けるべく、実習型授業が設けられており、その一環としてサービス指向の設計を学ぶ授業が開講されている。この授業においては、サービス指向設計について身をもって学ぶため、PBL で開発したシステムの拡張機能を検討し、それらの機能を SOA の考え方にに基づき、設計する演習を行った。設計の際には、モデルベース開発の教科書を参考にし、そのプロセスに従ってサービス指向設計を行った[7]。

次節では、上述した演習で私たちが考えた拡張機能と行った設計について、設計時に作成したアクティビティ図、ユースケース図、クラス図を基に説明する。

4.2. 設計の詳細

最初に現行の業務を分析し、図 4.1 に示すように現行のアクティビティ図を作成した。

現行の業務では、注文を行う前に書籍情報の事前調査が必要となり、また、注文を行う度に顧客情報の入力が必要であるため手間がかかるなどの非効率な面がある。私たちは、上述した問題を解決するために書籍検索機能と顧客登録機能を拡張機能として考えた。

書籍検索機能とは、キーワード入力により書籍の検索ができる機能である。この機能を利用することで出版社や ISBN などの細かい情報の事前調査や入力が必要となる。顧客登録機能とは、顧客情報の登録を行う機能である。この機能を利用することで、登録後の注文に対して顧客情報の入力が不要となる。これらの拡張を反映したアクティビティ図を図 4.2 に示す。

次に、ユースケース図とユースケースシナリオを作成した。ユースケース図を図 4.3 に示す。ここで私たちは、書籍検索機能と顧客登録機能を新たにシステムに含めて構築するのではなく、運営会社のシステム機能を利用することにした。図 4.3 により、取次店検索システムと顧客情報管理システムが本システムの外側にあり、サービスとして利用おり、ユースケースレベルでサービス指向の考え方を取り込んでいることが分かる。

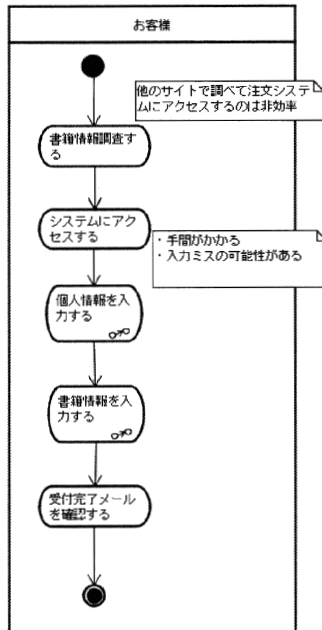


図 4.1 アクティビティ図(AsIs)

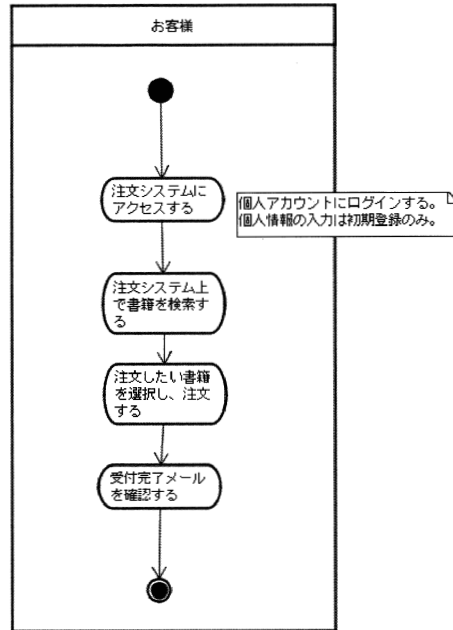


図 4.2 アクティビティ図(ToBe)

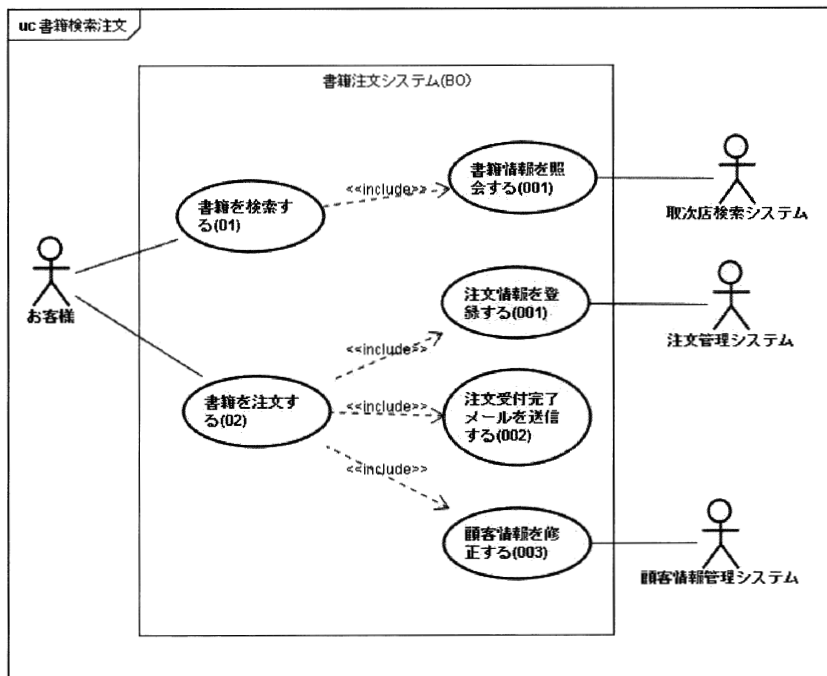


図 4.3 ユースケース図(サービス指向)

次に、ユースケースシナリオの入出力情報を基にオブジェクト図、オブジェクト図を基に分析クラス図を作成し、パッケージの切り分けを行った。それで、最終的に図 4.4 に示すようなクラス図を作成することができた。図 4.4 により、データと処理機能が明確に区別されており、パッケージとしても独立していることが分かる。

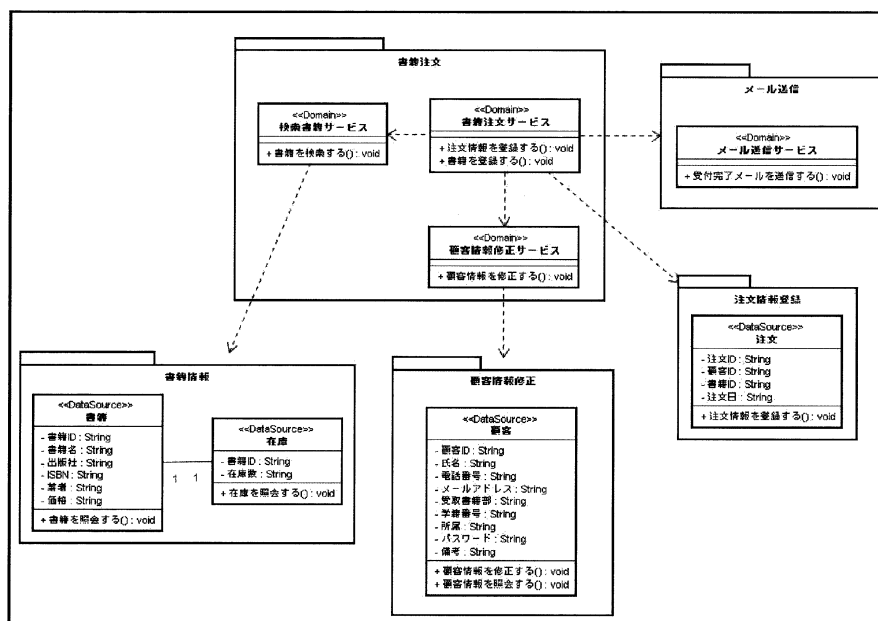


図 4.4 クラス図(サービス指向)

5. 比較評価

本章では、独自手法による設計とサービス指向設計を比較・評価し、考察を行う。

5.1. ユースケース図の比較考察

図 3.1 と図 4.3 を比較してわかるように、サービス指向の考え方に従って作成したユースケース図においては書籍注文システムが取次店検索システム、注文管理システム、顧客情報管理システムとがそれぞれシステムとして独立しており、独立性を保っている。これによって、一部のシステムの機能が変わった際も他のシステムにおける機能への影響を最小限に抑えることができる。システムを拡張する際にも、内部的に新しい機能を組込むのではなく、外部から新たなサービスを読み込む方式にすることで比較的容易に拡張が可能となる。また、取次店検索システムを使うことでソフトウェアの再利用が行われている。また、各処理機能をサービスとしてまとめたことによってユースケースの数が少なくなり、可視性も向上している。

5.2. クラス図の比較考察

独自設計(図 3.2)とサービス指向設計(図 4.4)の決定的な違いとして挙げられるのはパッケージのまとめ方である。独自設計のクラス図は書籍注文パッケージにすべての機能が入っているのに対して、サービス指向設計のクラス図は細かい粒度によりパッケージとしてまとめられている。パッケージやクラスの数、クラス間の依存度などの視点から比較したものを表 5.1 に示す。以下、これらを基に可視性、柔軟性、拡張性、再利用性について考察を述べる。

表 5.1 独自設計とサービス指向設計の比較

比較項目	独自	サービス指向
パッケージの数	1	5
1パッケージあたり最大クラス数	11	3
1パッケージあたりの平均クラスの数	11	1.6
意味のある機能単位でパッケージ化されている	△	○
データと処理が区別されている	△	○
一つのクラスに対して使用するクラスの数(最大数)	4	1

(1) 可視性

表 5.1 に示すように独自手法によって作成したクラス図は、パッケージとしてのまとまりが少なく、一つのパッケージ内に存在するクラスの数が多い。また、クラス図を見てもわかるようにそれぞれのクラスの結びつきが複雑である。それに加えて処理機能を持つそれぞれのクラスやデータのクラスが明示的にわけられていないため、クラスの役割や関連がわかりにくいものとなっている。

一方、サービス指向設計によるクラス図は、処理を行うクラスと、データクラスとが明示的に分けられており、細かい粒度によりパッケージとしてまとめられている。また、それぞれのパッケージが一定の意味のあるサービスの単位としてまとめられており、それぞれのパッケージの持つ処理の役割が明確なものとなっている。各処理機能をまとまりとしてクラス図全体を見ることができる。可視性が高いと言える。

(2) 柔軟性

図 3.2 や表 5.1 を見てもわかるように独自設計において作成したクラス図は、1つのクラスを複数のクラスが利用している場合が多く、クラス間の関連が強い。仮に顧客情報クラスを修正しようとした場合は、それを使用している他の4クラスの修正が必要となる。

一方、サービス指向設計によるクラス図は、依存度が低いため、一つのクラスを修正した場合の影響度が少ない。このことから、クラス図(実装)レベルにおいてもサービス指向設計は柔軟性を持つ設計手法であると言える。

(3) 拡張性

上記で述べたように独自設計においては、それぞれのクラスの関連が強いため、新たに新機能を追加する際にもより多くの修正が必要となる。今回作成した注文システムは、書籍のみを取り扱うことを想定としているが、今後の展開において書籍以外の商品をシステム上で取り扱いたい場合などに新たにクラスやパッケージを追加する際にも既存のクラスの関連を慎重に考慮しなければならない。

一方、サービス指向に基づくクラス図は、別の検索サービスのパッケージを追加し、検索注文のクラスを修正することで拡張が可能である。

(4) 再利用性

独自設計によるクラス図は、パッケージの粒度が大きく、またそれぞれの処理が関連し合っており、汎用的に利用できるものとしてまとめられていないと言える。再利用は不可能ではないが汎用的でない面がある。

サービス指向による設計においては、パッケージの粒度が小さく、機能ごとにわけられている。特にメール送信、顧客情報登録サービスに関しては書籍以外の注文システムを構築する際にも再利用が可能であると考えられ、再利用性が高い設計が行われていると言える。また既に、書籍検索機能に関しては既存の外部サービスを使用しており再利用を行っている。

以上の観点から設計の比較の評価をまとめたものを表 5.2 に示す。考察の結果、総合的にサービス指向による設計の方が優れているという結論となった。

表 5.2 評価のまとめ

評価項目	独自	サービス指向
クラス図がわかりやすい	△	◎
データ構造や処理の変更が容易である	×	○
新たな機能(サービス)の追加が容易である	×	○
パッケージの再利用ができるか	△	○

6. まとめ

本論文では筑波大学大学院にて行われている PBL 型システム開発におけるサービス指向設計の試みを紹介し、PBL 型システム開発において行った独自の設計と新たにサービス指向に基づいて設計を行ったものを比較・考察し、それぞれの設計に対して可視性、柔軟性、拡張性、再利用性などの観点から評価を行った。その結果、独自設計よりもサービス指向に基づく設計の方が可視性、柔軟性、拡張性、柔軟性の観点から優れているという結論が得られ、SOA の有効性が確認できた。

参考文献

- [1] 高度 IT 人材育成のための実践的ソフトウェア開発専修プログラムホームページ：
<http://www.cs.tsukuba.ac.jp/ITsoft/>
- [2] 日本 PBL 研究所ホームページ：<http://www.pbl-japan.com/>
- [3] Dirk Krafzig, Kar Banke, Dirk Slama, 山下真澄(翻訳)：SOA 大全，日経 BP
- [4] 鶴保征城，駒谷昇一：ずっと受けたかったソフトウェアエンジニアリングの授業(1)，株式会社翔泳社
- [5] 鶴保征城，駒谷昇一：ずっと受けたかったソフトウェアエンジニアリングの授業(2)，株式会社翔泳社
- [6] UML specification：<http://www.omg.org/spec/UML/2.1.2/Infrastructure/PDF>
- [7] 桐越信一，長瀬嘉秀：現場の UML，株式会社ソーテック社