

オフショア・ソフトウェア開発を想定した 日中共同作業実験の提案と評価

玄 麗花[†] 由井 蘭隆也[†]

オフショア・ソフトウェア開発では仕様書の取扱いとコミュニケーションが課題とされる。そこで、オフショア開発を想定した日中共同作業実験を考案し、評価実験を行った。その実験では国外開発の模擬として中国人グループ、国内開発の模擬として日本人グループ、そして、協力開発として日中グループを参加者とする。その実験結果をもとに成果物と共同作業の特徴を検討した。

Proposal and Evaluation of Laboratory Experiments for understanding Offshore Software Development

Lihua Xuan [†] and Takaya Yuizonon[†]

Issues of offshore software development are how to treat specification and its communication. To consider such a development in groupware technology, we propose experiments in laboratory settings. In the experiment method, Chinese group, Japanese group and Japanese-Chinese group correspond to offshore development, domestic development, and cooperative development, respectively. The results indicated that collaboration was important for system production in the experiments.

1. はじめに

近年、日本と中国との企業取引や民間交流が幅広く行われている。その取引の中で、知識労働が必要な代表的取引としてソフトウェア開発を上げることができる。日本のオフショア・ソフトウェア開発は1980年代にコスト削減を動機として開始されてきたが、今は、委託側・受託側双方の信頼関係を重視した上で、ソフトウェア開発のプロセス全体を管理運営する段階にきているとされる[1]。日本におけるオフショア開発は委託先として中国が、中国におけるオフショア開発は受注先としては日本が共に一番の割合である。よって、オフショア開発先としてインドやベトナムなどが台頭しつつあるが、以前として、日本が中国で行うオフショア・ソフトウェア開発は重要といえる[2]。

このようなソフトウェア開発を全体的に理解するためには、ソフトウェア・プロセスを観察して理解することや日誌による調査などが行われる[3]。しかしながら、オフショア・ソフトウェア開発の新たな形式、例えば、日中協力型の形式を調べることは容易ではない。そこで、研究室で行えて、かつ、オフショア・ソフトウェア開発の共同作業を取り扱える共同作業実験を検討することにした。

本報告では、オフショア・ソフトウェア開発で課題とされる部分に注目し、その開発の課題点を調べるための研究室実験を提案する。その実験方法を用いて、オフショア開発、国内開発、日中協力型開発を検討する。

2. オフショア・ソフトウェア開発を想定した共同作業

2.1 共同作業内容

オフショア・ソフトウェア開発の問題点を想定した研究室内で行える日中共同作業を検討する。オフショア開発の問題点に関する企業アンケート結果によると、日本企業の場合はコミュニケーションを問題とする企業が約2割、海外企業の場合、仕様書確定・変更を問題とする企業が約3割、コミュニケーションを問題とする企業が約2割である[4]。よって、提案する共同作業は仕様書とコミュニケーションに関するものとした。一方、仕様書にもとづくシステム開発であるが、数時間程度で行う研究室実験で行うためにはプログラム開発は困難である。また、プログラムによるシステム開発（例えば、Web サービス開発）を行える人材を揃えることにも困難が予想される。そこで、プログラミング開発の代わりにシステム図を作成することで代替することを考えた。その図の作成にはラベルや矢印を用いることとする。

オフショアを想定した共同作業の概観は図1に示す通りであり、委託側と受託側はビデオ会議システムなどの通信手段を用いてしかコミュニケーションを取れない状態

[†] 北陸先端科学技術大学院大学 知識科学研究科
School of Knowledge Science, Japan Advanced Institute of Science and Technology

にする。受託側のチームは3人構成とし、1人をチームリーダー、2人はプログラムの役割を演ずる。ただし、ここではプログラムは行わずに作業を行うため、プログラマの代わりにワーカと呼ぶ。リーダーは直接、作業に手を出さないが、ワーカの作業をチェックし、作業に対する指示を出すことができる。

具体的に行う作業は、委託側から発注された想定された仕様書をもとにシステム図を作成することである。このシステム図の作成は紙や鉛筆を用いてもよいが、システム図をリーダーが確認できるように画面共有機能をもったソフトウェアと図作成機能をもつソフトウェアを使うこととする。よって、最終成果としての図をコンピュータ上で作成したものとする。また、オフショア開発の問題点として仕様書の変更があげられている。そこで、実験の途中で仕様書の変更を行う。その変更にもとづいて受託側はシステム図を修正、改良する必要がある。

最後に、受託側のチーム構成について説明する。チーム構成としては次の3パターンを考え、これらパターンを比較することによって、オフショア開発における特徴を抽出することを期待する。パターン一つ目は現状のオフショア開発を模擬する従来型である。受託側全員を中国人にする（中国 G とも呼ぶ）。リーダーは日本語が上手な人にし、ワーカは日本語の文章は読めても、会話は流暢にできない人にする。海外のオフショア開発を比較するために、パターン二つ目は従来の国内開発を模擬する国内型である。受託側を全員日本人にする（日本 G とも呼ぶ）。これはオフショア開発を行わない従来の形態である。パターン三つ目は日本人と中国の共勤が進んだ状態である共勤型である。受託側はリーダーを日本人にし、ワーカは中国人にする（日中 G とも呼ぶ）。

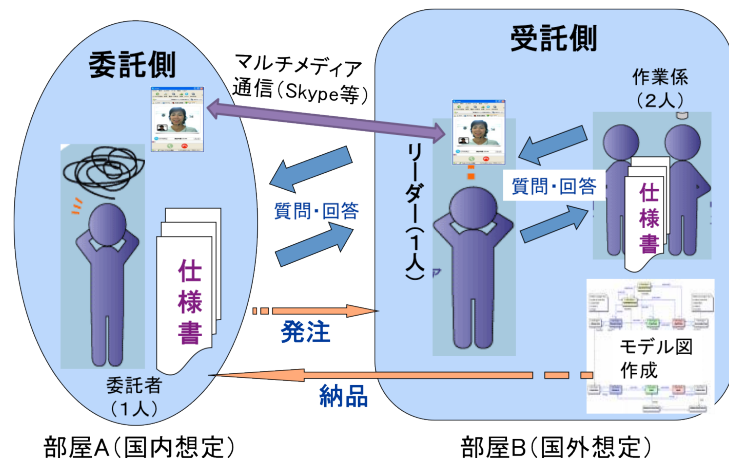


図1 オフショアを想定した共同作業の概観

2.2 仕様書

実験で使う仕様書の問題対象はプログラム設計技法の比較用に作られた「酒類販売会社の倉庫システム」である[5]。そして、仕様書の構成は、日本情報システム・ユーザー協会による「要求仕様定義ガイドライン」[6]を参考にした。そして、重要な仕様の文言記述部分には大木らによる日本語による説明記述[7]を参考にした。その日本語記述は、並行オブジェクト処理を意識して記述されているとともに、プログラム設計技法に依存した記号記述は行われていない。よって、現在のオブジェクト指向開発における仕様書として相応しい。その仕様書の一部を図2に示す。

- 1、はじめに
 - 1.1、目的：在庫管理システムのモデル化
 - 1.2、範囲：酒屋在庫管理
 - 1.3、用語の定義：
 - 内蔵品：倉庫に保管されている品目ごとの商品である。
 - コンテナ：積荷品を混載しているコンテナである。
 - 積荷品票：積荷品の1要素で、積荷品とコンテナの対応を記録している。
- 2、要求と仕様
 - 2.1 入庫受付係
 - 2.1.1、コンテナが到着すると、
 - 2.1.1.1、コンテナを作る。
 - 2.1.1.2、コンテナに格納している銘柄ごとに、
 - a、内蔵品に入庫数量を知らせる。
 - b、積荷品票を作る。
 - 2.2 出庫受付係
 - 2.2.1、出庫依頼が来ると、内蔵品に出庫依頼を知らせる。
 - 2.3、内蔵品
 - 2.3.1、入庫受付係より入庫数を受け取ると、
 - 2.3.1.1、在庫数を在庫数+入庫数とし、在庫不足に在庫数を知らせる。
 - 2.3.1.2、在庫不足が解消されたために、出庫する必要があるが生じたならば、
 - a、在庫数を在庫数-出庫数とする。
 - b、積荷品票に出庫指示を送る。
 - c、未出庫分があるならば、再度、積荷品票に出庫指示を送る。
- 2.4、在庫不足

図2 倉庫システムの仕様書の一部（下線は変更部分）

最初に、システムの目的や用語の定義が記述され、それ以降は、要求と仕様が述べられる。「入庫受付係」、「出庫受付係」、「内蔵品」、「在庫不足」といったオブジェクト

(またはモジュール)が定義され、それらオブジェクトに対して起こるイベントによって振る舞いが記述されている。下線部は、共同作業の途中に行われる仕様書変更部分である。

3. 提案共同作業の評価実験

3.1 実験内容

提案した共同作業を用いてオフショア・ソフトウェア開発を検討するための実験を行った。実験は、3パターン(従来型、国内型、協力型)、2グループずつの計6回行った。受託側1グループあたり、リーダー1人、ワーカ2人の計3人で構成される。実験の参加者は本学の大学院生18名であり、日本人8名、中国人10名であった。中国人の日本語を勉強した年数は、中国Gのリーダーは8年以上、ワーカは平均2年、日中Gのワーカは平均3年であった。ほとんどの日本滞在年数は平均3年未満であった。委託側は実験者の1人である本学教員が務め、全6回の実験を通して変わらなかった。その本学教員は情報工学科出身でソフトウェア工学を勉強した経験があるものであった。

実験手順について述べる。まず実験参加者に対して実験の内容と共同作業に用いるシステムの使用方法を15分程度で説明する。その説明の際、ソフトウェア工学の教科書[8]を参考にして作成した仕様書とシステム図を用いた。その後、実験に入る。実験時間は合計で50分である。この時間は実験者2名による予備実験においてシステム図作成を行ったところ、2人とも計算機を使って約50分で完了できたためである。当初、実験参加者には、実験時間は30分と告げ、倉庫システムの仕様書を配る。また、各人にA3の紙と筆記用具を渡し、自由に使ってもらった。そして、実験時間が20分経つと、実験者の1人である委託側は仕様書の変更を告げ、新たに仕様書を受託側に配る。

実験終了後、実験参加者全員にアンケートや仕様書のわからなかった部分の調査を行った。実験参加者へのアンケート評価は主に5段階評価で行い、評価が低い場合、その理由を自由記述させた。また、実験で配布した仕様書にわからなかった部分に線引かせると共に、その理由や意見を記述させた。

3.2 実験環境

実験は北陸先端科学技術大学院大学の知識科学研究科Ⅲ棟6階の研究室にある研究員用ブースと教員室間で行われた。この二つの場所同士のコミュニケーションには、インターネット電話として広く使われるSkypeを使い、委託者と受託側リーダーとのやりとり、主に、質問・回答に用いられた。受託側はシステム図を作成するために、分散協調型KJ法支援ソフトKUSANAGIを用いた[9]。このKUSANAGIは、分散協調型KJ法を複数人で行うための画面共有機能やラベル出し機能をもつと共に、矢印

機能が追加されている。図3に受託側環境の様子を示す。参加者それぞれがPCを用いた作業を行うことができた。また、リーダー用PCにはSkypeも動いていた。

一方、実験状況や音声会話を記録するためにビデオ撮影を行った。また、画面共有ソフトであるVNCと画面キャプチャソフトを連携することにより、システム図を画面データとして定期的にファイル保存した。



図3 受託側環境の様子(左側2人はワーカ、右側1人はリーダー)

4. 実験結果と考察

4.1 実験結果

オフショア・ソフトウェア開発を想定した共同作業実験の結果として、仕様書をもとに作成されたシステム図が6つのグループより得られた。その図の評価とグループ成員の知識背景をまとめた結果が表1である。

システム図を評価するために仕様書得点という点数付けを行った。まず、最終的な仕様書の「2. 要求と仕様」に記述された32項目がシステム図に表現されているかどうか、1つずつチェックする。各項目に、できている項目に○、まあまあできている項目に△、できていない項目に×をつける作業を行った。この作業は、委託側を担当した教員Aと実験者である大学院生Bによって行われた。ここで、○を1点、△を0.5点、×を0点として足したものを仕様書得点とした。評価者Aと評価者Bとの間の相関を調べると、項目ごとのチェックでは0.75、システム図ごとの得点では0.99と高い相関が得られた。

また、その表にはシステム図ごとのラベル数、矢印数を示した。また、各ラベルと各矢印については、仕様書を反映しているかどうかという観点から、できている項目

に○, まあまあできている項目に△, できていない項目に×をつける作業を行い, 仕様書得点と同等な方法で点数付けを行い, それぞれラベル得点, 矢印得点を付けた.

グループ成員の知識は実験参加者の自己判断で行われ, 情報の知識をもっているかどうかで Yes と答えた人数, そして, 仕様書の学習を 5 段階評価で申告した結果の平均値を示している. 値 5 は, とても学習した, 値 1 は, とても学習していないに対応している.

評価結果より, 約 7 割の得点をとっている中国 G2 を筆頭に, 日本 G2, 日中 G2 までは 6 割の得点をとっていた. 以下, 日本 G1 は約 5 割, 中国 G1 は約 4 割, そして, 日中 G2 は約 1 割の得点しか取れなかった. 評点が 6 割を越えたグループはラベル数と矢印数が多いことがわかる. 一方, 得点が低い中国 G1 はラベル数と矢印数が少ないことがわかる. そして, 日中 G1 は作られたラベルと矢印の正確さも劣ることがわかる. 評価が高かった中国 G2 によるシステム図を図 4 に, 低かった日中 G1 によるシステム図を図 5 に示す.

この結果を, 各グループ成員の知識背景と見比べてみたが, 本実験にとって有利と考えられる知識背景を持つ中国 G2 と日本 G1 では差が見られた. 同様に不利と思われる中国 G1, 日中 G1 は悪い成績であったが, 日中 G2 は可となる成績を上げていた. 以上より, ソフトウェアに関する勉強経験を持っていると答えたグループが当初はよい成果を出す予想していたが, 必ずしもよい成果物を作るわけではないという結果になった.

表 1 システム図の評価とグループ成員 3 人の知識背景

	中国G1	中国G2	日本G1	日本G2	日中G1	日中G2
仕様書得点(32点満点)	12	23	17	21	4	20
ラベル数	14	24	24	27	17	26
ラベル得点	11	23	20	25	6	21
矢印数	8	21	21	20	11	27
矢印得点	8	21	14	19	4	21
情報の知識	1人	全員	全員	全員	全員	2人
仕様書の学習	1.7	4.0	3.7	2.7	1.7	1.7

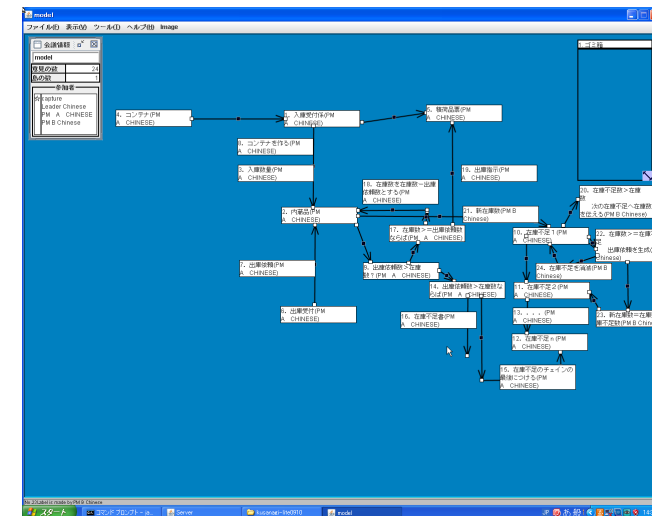


図 4 評価が高かったシステム図 (中国 G2 による)

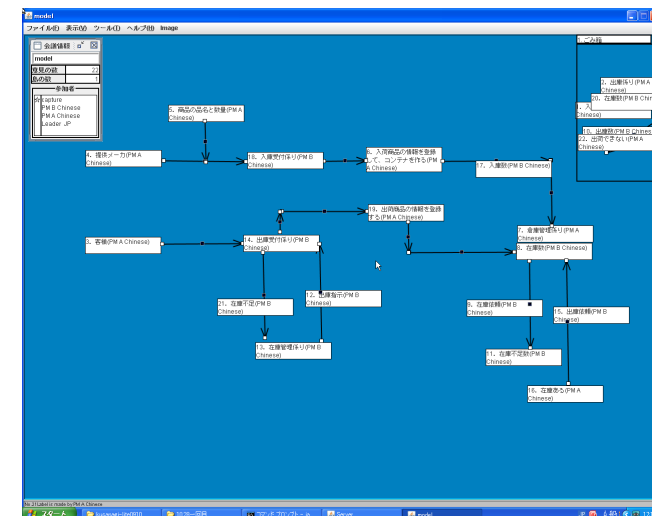


図 5 評価が低かったシステム図 (日中 G1 による)

4.2 アンケート結果

実験終了後, 実験参加者に行ったアンケート調査結果について述べる.

そのアンケート結果において、5段階評価で行ったアンケートの結果を表2に示す。最も高い評価を5、最も低い評価を1とした。その中で、Q8とQ9はワークのみに尋ねた。また、日本人参加者と中国人参加者の評価値を一元配置分散分析において有意差が出た場合には*印を表の右側に付けた。

表2 5段階評価アンケートのまとめ

質問項目	5段階評価
Q1.実験内容の分かりやすさ	3.3
Q2.興味を持って取り組めたか	3.8
Q3.仕様書のモデルに対するイメージ	3.1
Q4.仕様書のわかりやすさ	2.6
Q5.システム図の満足度	2.5*
Q6.共同作業はうまくいったか	3.7
Q7.グループ内コミュニケーションはとれたか	3.6
Q8.聞きたいことはリーダーに伝わったか	4.6*
Q9.リーダーが言うことを理解したか	4.4*
Q10.委託側とのコミュニケーション	3.3*
Q11.聞きたいことは委託側に伝わったか	4.1
Q12.委託側が言うことを理解したか	3.7*
Q13.異文化の壁を感じるか?	2.6

(注)Q8とQ9:作業係のみに質問

*:ANOVAtestで日本人と中国人間で $p < 0.05$ (参考)

表2より、全体的な実験参加者の評価をまとめると、実験自身は興味をもって取り組んでいる、共同作業はうまくいき、ある程度、グループ内コミュニケーションはとれ、聞きたいことは委託側に伝わったと感じているという結果になった。

次に、日本人と中国人間での評価において差があった点について述べる。システム図について満足したかどうか尋ねたところ、日本人と比べて中国人は満足しているという評価結果になった。仕様書得点がある程度ある中国G2、日中G2の中国人参加者であれば問題点はないが、得点が低い中国G1、日中G1の中国人参加者も満足していると答えているという結果になった。従って、システム開発において委託側は進捗評価をしっかりと行い、その上で開発を進める必要がある。

次に、作業者のみへの質問であるが、グループ内リーダーへの質問と応答については、日本人はうまくいったが、中国人はどちらともいえないという結果になった。これは、中国人同士のみの場合は、母国語の中国語で話せるが、日中協力型の場合、日

本人リーダーと日本語でコミュニケーションをとる必要があり、気軽に行えないためと考えられる。また、委託側とのコミュニケーション、委託側の回答を理解したかどうかについても、日本人はうまくいったと考えているのに対して、中国人はどちらともいえないと考えているということがわかった。従って、共同作業のコミュニケーション意識には言葉の違いが影響していることが伺える。

最後に、実験終了後に実験参加者を書いてもらった仕様書に対する疑問点についてまとめる。その結果、全32項目中12項目について疑問が表明されていた。日本人と中国人の共通問題点は5項目であった。具体的には、表現が同じ言葉の繰り返し(例:コンテナ、在庫不足)、表現が曖昧な部分(例:未出庫分があるならば、在庫不足が解消されたため、出庫が必要ならば)、カタカタと専門用語との区別(例:チェイン)であった。そして、中国人参加者には7項目の疑問点が追加される。具体的には、言葉の定義(例:コンテナ、内蔵品、在庫不足)、表現が曖昧なところ(例:次の在庫不足、自分自身)、出力先が明示されていないところ(例:在庫不足書を出力する、出庫依頼を生成するなど)であった。

以上より、中国人は日本人より言語的に不利な点があり、日本人側はコミュニケーションに注意が必要である。

4.3 共同作業の様子

実験の様子を撮ったビデオを観察して、参加者の協力状態とコミュニケーションの様子をまとめたものが表3となる。

ワークを中国人に注目すると、自己主張が多いという様子が目立ち、文化習慣的には日本人は同質化を好むことに対して、中国人は個人主義であることが現れた可能性がある。しかしながら、得点が良かった中国人G2は全員が協力して作業を行っていたことがわかった。

ソフトウェア仕様書に関する学習経験が少ないグループは中国G1、日中G1、日中G2である。その中で、6割の得点を確保している日中G2は、三人の間でコミュニケーションをとると共に、作業分担はあったが計算機画面上で共有した作業を行っていた。一方、中国G1、日中G1のワークは作業分担を行っているが、別々にシステム図を作成し、後から組合せようとしていることがわかった。

以上より、得点が低い班は作業の共有が少ないと同時に自己主張を行うワークがいる班、つまり作業を協調して行えていない状況がわかった。このことから提案した共同作業実験においては、共同作業を協力して行えるかどうかが重要であると考えられる。

今回、紙によってバラバラな共同作業を許したが、グループウェア等の共同作業環境のみしか作業環境を提供しないことによって、共同作業を強制するという方向性がある。しかしながら、この結果をソフトウェア開発に応用できるかどうかの判断は拙速であり、ソフトウェア開発プロセスにおける共同作業の有効性を調査してから検討する必要がある。

表3 ビデオ観察のまとめ

	協力の状態	コミュニケーション
中国 G1	仕様書の分担があった。ワーカーの一人は先に紙ベースで作った後、システムで作り、もう一人は直接システムで作った。	三人の間に意見交換があった。特に一人のワーカーは自己主張が多かった。
中国 G2	仕様書の分担はしなかった。ワーカーの一人は紙ベースで作り、もう一人はそれを参考にシステムで作った。	一行一行に三人が一緒に分析し、盛り上がるほど意見を交換した。
日本 G1	仕様書の分担はしなかった。ただし、リーダーの指示に従って、その場その場でワーカーが自然に分担した。三人とも紙を使わなかった。	三人の間にコミュニケーションはあったが、質的な意見交換が少なかった。リーダーが仕様書を一行一行読んで、敬語を使って、ワーカーに指示を出した。
日本 G2	仕様書の分担があった。ワーカーの一人は先に紙ベースで設計してからシステムで作り、もう一人は直接システムで作った。	一人のワーカーは黙った状態で、もう一人のワーカーは活発で、リーダーとよく意見交換をした。
日中 G1	仕様書の分担があった。2人のワーカーは別々に紙に作った後、システムで作った。	リーダーとワーカー間は意見交換があったが、ワーカー間のコミュニケーションは少なかった。一人のワーカーは自己主張が多く、もう一人はネガティブな話が多かった。
日中 G2	仕様書の分担があった。2人のワーカーは直接システムで作った。	三人の間にコミュニケーションがあった。一人のワーカーは自己主張が多く、もう一人は質的な意見交換が少なかった。

5. おわりに

本研究では、日中オフショア・ソフトウェア開発を想定した日中共同作業を考案し、実験を行った。予め用意した仕様書を用いて、プログラミング開発の代わりにシステム図の共同作成を行う。そして、グループの形態としては従来型、国内型、協力型という三パターン検討した。

この提案実験では、(1) 共同作業をうまく行えているかどうか作業結果に重要な役割を果たすこと、(2) 中国人が含まれるグループでは、日本語によるコミュニケー

ションに壁があることや自己主張が多いワーカーが生じ易いという傾向、(3) 中国人ワーカーはシステム図の得点評価に関係なく作業に満足しているという傾向が得られた。

今後は、共同作業を引き出すためのインタフェースや文化的習慣を考慮したインタフェースなどが検討課題である。一方、ソフトウェア開発プロセスにおいて共同作業の役割を調査することも課題である。

参考文献

- [1] 辻洋, 守安隆, 盛忠起: オフショア・ソフトウェア開発の進化と技術者の経験知, 情報処理, Vol.49, No.5, pp.551-557(2008).
- [2] 総務省: オフショアリングの進展とその影響に関する調査研究報告書 (2007).
- [3] Perry, D. E., Staudenmayer, N.A., and Votta, Jr. L.G.: ソフトウェア開発における時間利用についての理解とその改善, ソフトウェアプロセスのトレンド, pp.175-209, 海文堂 (1997).
- [4] 中原俊正, 藤野博之: オフショアプロジェクトにおける UML の適用, プロジェクトマネジメント学会誌, Vol.10, No.6 (2008).
- [5] 山崎利治: 共通問題によるプログラム設計技法解説, 情報処理, Vol.25, No.9, pp.934(1984).
- [6] 日本情報システム・ユーザー協会: 要求仕様定義ガイドライン (UVC), (2007).
- [7] 大木優, 二村良彦, 竹内彰一, 宮崎敏彦, 古川康一: 論理型並列プログラミング言語 Concurrent Prolog[®]による在庫管理システムの記述, 情報処理, Vol.26, No.5, pp.470-476(1985).
- [8] 中所武司: ソフトウェア工学第2版, pp.42-46, 朝倉書店 (2004).
- [9] 由井菌隆也, 宗森純: 研究グループの知識創造活動を支援する GUNGEN-SECI の表出化と連結化, 情報処理学会論文誌, Vol.48, No.1, pp. 30-42(2007).