

## 再利用性を重視したオブジェクト指向設計技法

中條 貴幸<sup>1</sup> 松井 雅行<sup>2</sup> 吉沢 比呂志<sup>3</sup>  
古賀 恵子<sup>4</sup> 尾崎 貴則<sup>5</sup> 倉山 忠<sup>6</sup>

<sup>1</sup>日本電気アイ・マイコンシステム(株) <sup>2</sup>アンリツ(株)

<sup>3</sup>(株)インテック <sup>4</sup>(株)日立製作所

<sup>5</sup>山武ハネウェル(株) <sup>6</sup>住友重機械工業(株)

オブジェクト指向設計技法の中でも、特に再利用性を重視したSPECという技法の検討結果を報告する。SPECは要求定義も含めて、オブジェクトの抽出、クラスの抽出、継承を用いたオブジェクトの抽出を行なう技法である。また、SPECの入力は非形式的なユーザ要求、出力は形式的な仕様である。そして、出力された仕様を基にインプリメンテーションが行なわれる。本報告では、SPECによる具体的なオブジェクト指向設計方法を示すとともに、再利用性に関する検証結果を報告する。

## OBJECT ORIENTED APPROACH ENRICHING DESIGN REUSABILITY

TAKAYUKI CHUJO<sup>1</sup> MASAYUKI MATSU<sup>2</sup> HOROSHI YOSHIZAWA<sup>3</sup>  
KEIKO KOGA<sup>4</sup> TAKANORI OZAKI<sup>5</sup> TADASHI KURAYAMA<sup>6</sup>

<sup>1</sup>NEC IC MICRO COMPUTER SYSTEM,LTD <sup>2</sup>ANRITSU CORPORATION

<sup>3</sup>INTEC INC. <sup>4</sup>HITACHI INC

<sup>5</sup>YAMATAKE HONEYWELL INC <sup>6</sup>SUMITOMO HEAVY INDUSTRIES,LTD.

The authors discuss Object-Oriented Design(OOD) method "SPEC" that especially places much phasis on reuse than other OOD methods do. "SPEC" gives object definitions, class definiton and object definition using class inheritances that contains definition of user's requirements. "SPEC" changes user's informal requirements into formal specifications. And we implemente the formal specifications. This paper report concrete OOD technique, and verification of reusability by "SPEC".

## 1 はじめに

筆者らは産学共同研究として日科技連ソフトウェア品質管理研究会平成元年および2年度において、オブジェクト指向の考え方とその技法が再利用性向上にどの程度役立つかを見極めるべく実践的に検討し、評価した。そこで検討結果を報告する。

実験方法は、あるシステムをオブジェクト指向設計技法で設計して得た結果を、他の類似システム(注1)の設計に再利用し、変更必要な箇所とその量を分析検討した。クラス定義を行い、その再利用性を、クラス変数など幾つかの再利用要因に着目して整理した。

実験に用いられたオブジェクト設計技法はSPEC(SPC WG-2G's Entity Based Design Creator)と呼ばれる技法である。まず、この技法を紹介し、次にSPECを用いて行なった実験を示し、最後に再利用性に関する定性・定量的な評価を述べる。

## 2 オブジェクト指向設計技法

技法の説明の前に、これまでのオブジェクト指向設計技法の代表例をいくつか紹介する。

(a) Boochのオブジェクト指向開発[1], [2]

システムの分解作業を、オブジェクトの概念に基づいて行なう。オブジェクト抽出時の判断基準（名詞による）などを定めている。

(b) F. Manfrediらのオブジェクト指向システム分析[3]

オブジェクトとその構造に注目してクラスの抽出を行なう。

複雑なオブジェクトは複数のクラスを用いて記述する。

クラス間の使用、被使用の関係を明確にする。

---

(注1) 本報告ではあるシステムと同一ドメインであるが別のシステムを類似システムと呼ぶことにする。

(c) 大林氏の概念による設計法(DMC)[4]

問題を分析し、問題文中の単語を整理する。主たる概念となる単語をインスタンスとして抽出し、インスタンス間の関係を図示する。

インスタンスの振舞い、データを定義する。

(d) P. Wardらの構造化分析・設計手法によるオブジェクト指向設計[5]

S A / S D を使ってオブジェクト指向設計をする。

データフロー図中のデータに注目してオブジェクトを抽出する。

E-R 図を記述し、Entityをデータオブジェクトとし、Relationshipをデータ間の関係として抽出する。

(e) S. Shlaer, S. J. Mellorらのオブジェクト指向システム分析[6]

実世界上の同じ特徴を持ち、同じルールに従っている事象を捜す。その特徴やルールに着目して、事象を抽象化し、オブジェクトを抽出する。更に、オブジェクトの属性やオブジェクト間の関係を抽出する。

## 3 オブジェクト指向設計技法SPEC

筆者らの提唱するオブジェクト指向設計技法SPECについて記述する。

### 3.1 SPECの基本的な考え方

SPECでは3.3節から3.5節のオブジェクト／クラスの抽出に3.2節の要求定義も含め、それらを総称して設計と定義している。

SPECは日本語などで非形式的に書かれたユーザ要求を入力とする。そして、設計すべきシステムとシステム外のエンティティとの間のメッセージやデータによる関係をE-R図で定義する。次に、要求定義で作成された文書を入力として、オブジェクトの抽出を行なう。このとき、類似システムのオブジェクト抽出結果が存在する場合はその文書も入力とし、ユーザ要求の類似点から、設計結果のオブジェクトの抽出結果を類推し、オブジェクト候補を決定することで、短時間のうちにより正確に設計を行うことを目的とする。

確なオブジェクトの抽出を行なうことができる。

また、設計例がある場合に文書を利用するだけではなく、より抽象的で一般的な「クラス」の抽出を行ない、オブジェクト指向の利点の1つである継承の機能を活用する。クラスが抽出されている場合、オブジェクトはクラスを用いて定義する。その場合、クラス抽出の時に作成した文書は、直接、現在抽出しているオブジェクトの文書として使用される。

最後に上記SPECによる設計結果からの「インプリメンテーション」についての説明をする。

以下、それらを順に説明する。

### 3.2 要求定義

SPECによる要求定義の目的は、非形式的なユーザ要求から改良型コンテキストダイアグラムとシステム仕様と呼ばれる形式的な文書を作成することである。

そのときに、SPECで設計された類似システムの設計結果がある場合、それを利用する。その方法は、エンティティの定義、エンティティとシステムの関係の定義、システムの振舞いの定義に注目し、これから設計しようとしているシステムと名前が似ている点、名前が違っても同じ振舞い、同じデータを表わしている点を探し出す。探し出した利用可能部分は、名前の簡単な置き換えやメッセージによる関係の追加・削除、データの追加・削除を行なって改造型コンテキストダイアグラムやシステム仕様を作成していく。そのため、全くの新規要求定義の時よりも、短時間に、正確に作成できる。

また、SPECで設計した結果がないときでも、要求定義は可能である。この場合は、システム外に存在する「もの」を整理し、まとめていく。システムの外部で、「もの」が人の間をどのように動き、それによって人はこれから設計しようとしているシステムにどのように

な要求をしてくるかを明確にし、改造型コンテキストダイアグラムを作成する。ユーザの要求を整理して、それをメッセージとして、その入出力、振舞いを定義してシステム仕様を作成する。

#### (1) 改良型コンテキストダイアグラム

改良型コンテキストダイアグラムを図1に例示する。これは、コンテキストダイアグラムを改良したE-R図である。コンテキストダイアグラムとの違いの1つは、図の中心に描かれたシステムに対し、擬人化したエンティティからどのようなメッセージが入力されるかを示していることである。もう1つの違いは、システムの外部で擬人化したエンティティやそれ以外のエンティティ間を実世界の「もの」がどのように流れているかを示していることである。

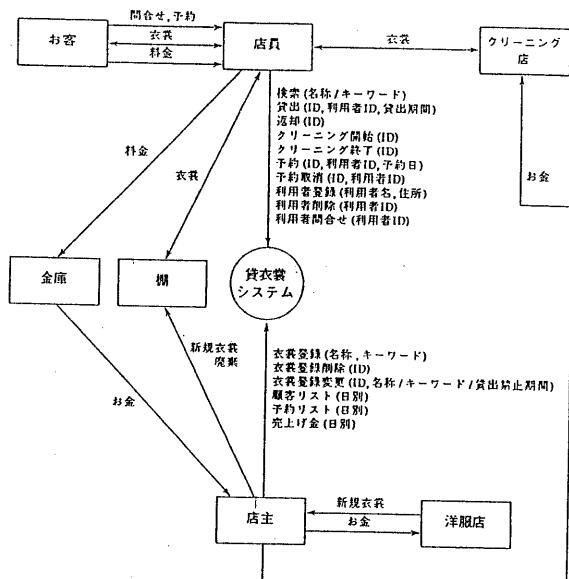


図1 改造型コンテキストダイアグラム  
(貸衣裳システム設計時の例)

#### (2) システム仕様

次に、与えられた非形式的なユーザ要求仕様と上記改造型コンテキストダイアグラムから、システムが外部から受けるメッセージの入出力と処理を日本語で記述する。これをシ

システム仕様と呼び、システムが外部からどのようなメッセージを受け、どのような振舞いをするかを明確にする。図2にシステム仕様の一部を例示する。

<b>1. 検索</b>
方向: 店員→貸衣裳システム
入力: 名称/キーワード
出力: (ID、名称、キーワード、貸出予約状況)/レンタル品なし
処理: 入力情報に基づいて、データテーブルを検索し、IDを見つける。そのIDより名称、キーワード、および貸出禁止期間や貸出日や予約日より貸出予約状況を出力する。
さらに、入力情報に基づいたデータテーブルを検索する。 見つからない場合はレンタル品なしで終了。
<b>2. 貸出</b>
方向: 店員→貸衣裳システム
入力: ID、利用者ID、貸出期間
出力: 貸出可/貸出不可/貸出中/予約中/利用者未登録
処理: IDよりデータテーブルを調べ、貸出禁止期間より貸出不可かどうかを出力。貸出日より貸出中かどうかを出力。予約日より予約中かどうかを出力。利用者クラスに利用者IDを問い合わせて、利用者未登録かどうかを出力。それらのどれにも相当しない場合、データテーブルの貸出者と貸出日を変更して貸出可を出力。 (以下省略)

図2 システム仕様（一部）  
(貸衣裳システム設計時の例)

### 3.3 オブジェクト抽出

SPECによるオブジェクト抽出のために要求定義で作成した改良型コンテキストダイアグラムとシステム仕様を入力文書とし、以下の手順で概念図とオブジェクト仕様を作成する。

概念図によってオブジェクト間の関係を明確にし、オブジェクト仕様によって1つ1つのオブジェクトのデータ構造やメソッドを明確にする。

また、類似システムのオブジェクト抽出の事例がある場合、そのシステムの概念図、オブジェクト仕様も入力して利用される。

#### (1) 概念図

SPECのオブジェクト抽出では、まず、オブジェクト候補を選択する。オブジェクト候補は、要求定義で作成済みの改良型コンテキストダイアグラムやシステム仕様に示されたデータやエンティティの中から代表的なものを選び出す。また、オブジェクトに特有なデータもエンティティとデータの関係から抽出する。

次に、同じく要求定義時の文書からメッセージとメッセージ入力時のシステムのメソッドをメッセージの種類や扱う内部データに着目して整理して、共通のデータとそれを扱うメソッドに分類していく。オブジェクト候補に整理したメソッドやデータを当てはめる。こうして、定義されたオブジェクトをオブジェクト間のメッセージによる関係で示したものが図3の概念図である。

類似システムの設計結果が存在する場合には、その概念図やオブジェクト仕様書を、オブジェクト候補の選択やオブジェクトの関係の抽出に利用する。利用するときは、データ構造、オブジェクト間の関係、オブジェクトのメソッドなどの名前や動作や入出力に共通点があるかどうかに着目する。

オブジェクト候補の選定やオブジェクト間の関係づけは最終的な概念図が書かれるまでには、一般に何度も試行錯誤する。設計結果の活用によって、試行錯誤が減少する。

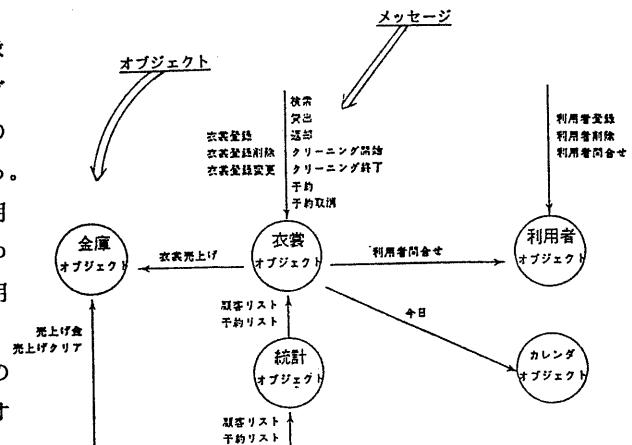


図3 概念図（貸衣裳システム設計時の例）

#### (2) オブジェクト仕様

概念図で示される1つ1つのオブジェクトについて、メッセージとその入出力、メソッド、メソッドのために必要なオブジェクト固

有のデータ構造を定義し、オブジェクト仕様（図4）を作成する。

概念図作成時に類似システムの設計結果を利用した場合は、オブジェクト仕様作成時にも概念図やオブジェクト仕様を利用する。

オブジェクト名： 貸衣裳オブジェクト	
[データ]	
	衣裳ID、衣裳名、色、柄、サイズ、購入価格、貸出者、貸出日、クリーニング日（開始と終了）、予約者、予約期間
[メッセージ]	
(1) 検索	入力： キーワードまたは衣裳名 出力： (衣裳ID、衣裳名、貸出可能日)/衣裳なし 処理： 入力されたキーワードまたは衣裳名にしたがって、衣裳データーベースを検索して衣裳IDを見つける。その衣裳IDの貸出日と予約日より、貸出可能日を求める。 さらに、同じキーワードまたは衣裳名で再検索する。 衣裳名なしになら終了する。
(2) 貸出	入力： 衣裳ID、利用者ID、貸出期間 出力： 貸出不可/貸出申込/予約申 処理： 衣裳IDより衣裳データーベースを調べ、本日より貸出期間内までに入力された利用者ID以外の予約があれば予約中を出力、貸出申込かクリーニング中だったら貸出中を返す。利用者がオブジェクトに利用者..... (以下省略)

図4 オブジェクト仕様（一部）  
(貸衣裳システム設計時の例)

### 3.4 クラス抽出

3.2要求定義や3.3オブジェクト抽出では、設計例を参考にする程度であった。ここでは、オブジェクトやその関係、オブジェクトのメソッドやデータ構造をより抽象的で一般的な表現とし、同一ドメインのシステムで共通に利用可能な「クラス」として定義する。クラスの定義により、現在設計している問題はもちろん、将来の類似システム設計時にも活用できるようにすることが目的である。

クラス抽出には「共通性による抽出」、「汎用性による抽出」の2つの着眼点がある。

このように定義されたクラス仕様を図5に例示する。

#### (1) 共通性による抽出

複数のSPECによる設計結果がある場合は、

「共通性」を1つの着眼点にしてクラスを定義することができる。つまり、複数の設計結果から、オブジェクトの基本的な概念の共通点、オブジェクトが受けるメッセージの共通点、オブジェクト内のデータ構造の共通点を見いだす。その共通点をより抽象化し、当該システムに固有の名称を一般的な名称に変更して、クラスとして定義する。

ただし、すべての設計結果に共通でなくとも同一ドメインの多くの類似システムで必要と判断されるオブジェクトはクラスとして定義する。

#### (2) 汎用性による抽出

SPECによる設計結果が1つしかなくてもクラスの抽出は可能である。概念図を抽象化したり、インスタンスやデータ、メッセージなどの名前が同一ドメインのどのシステムにも通用する抽象化した名前にできるかどうか、などを基準にしてクラスを定義する。

更に、SPECによる設計結果がない場合でも、現在設計しようとしているシステムだけでなく、同一ドメインのどんなシステムにも汎用的に利用できると考えるオブジェクトの抽出を行なう。基本的なオブジェクトの抽出の方法は3.3と同様であるが、オブジェクトを抽出するときにインスタンスではなく、より抽象的で一般的なクラスとして定義する。抽象化できない「もの」は、無理にクラス定義せず、3.5継承を用いたオブジェクトの抽出の時に定義する。

### 3.5 継承を用いたオブジェクト抽出

継承を用いたオブジェクト抽出は、3.3のオブジェクト抽出と同様に行なう。オブジェクト抽出との違いは既に作成済みのクラス仕様を利用するので、クラスの存在を意識して抽出を行なうことである。まず、どのようなクラス仕様があるか調査し、クラスを利用するオブジェクト候補を決定する。クラスを利用するときもオブジェクト間の関係は、3.3のオ

プロジェクトの抽出と同様、概念図で示す。さらに、概念図上のクラスを利用しているオブジェクトは図6に例示するようなオブジェクト内概念図を記述する。オブジェクト内概念図はオブジェクトとクラスの継承関係を明確にするために作成される。また、オブジェクト仕様は使用したクラス仕様に記述されていないデータやメソッドについてのみ記述する。

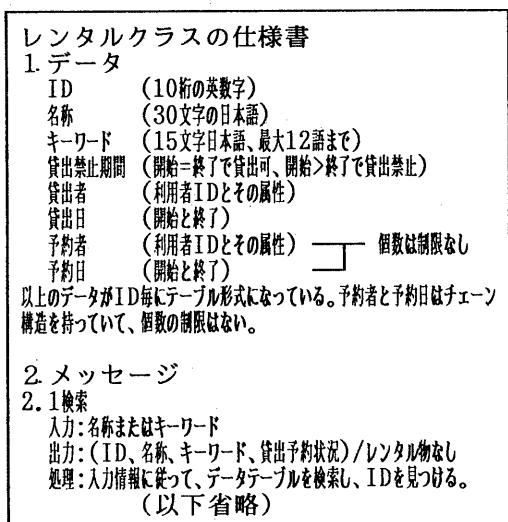


図5 クラス仕様  
(レンタルクラスの例)

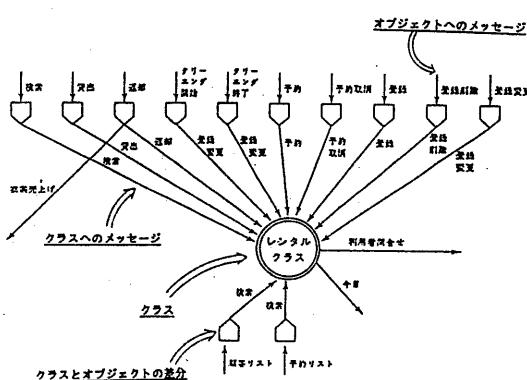


図6 オブジェクト内概念図  
(貸衣裳オブジェクトの例)

### 3.6 インプリメンテーション

SPECによる設計後、作成されたオブジェクト仕様、クラス仕様を用いてクラス、オブジェクトのインプリメンテーションを行なう。必要なデータ、メソッドはオブジェクト仕様、クラス仕様に記述されているのでそのままプログラミングする。しかし、この段階ではデータの名称の定義、メソッドが日本語で示されているだけなので、データ構造の正確な定義やメソッドの正確な記述をインプリメンテーションの段階で行なう必要がある。

## 4 SPECの実践

### 4.1 使用した問題

以下に紹介する3つの問題をSPECを用いて設計した。3つの問題は同一ドメインである「レンタルショップの管理システム」に関する類似問題である。以下に3つの問題の概略を示す。

#### (1) 図書館システムの問題

企業内の図書館における図書管理システムの設計を行なう。

求められている機能としては、

「照会」、「予約」、「貸出」、「返却」、「利用者登録・削除」、「貸出禁止処理」、「図書追加・削除」、「利用状況の統計処理」などである。

#### (2) ビデオレンタルシステムの問題

レンタルビデオショップのビデオの管理システムの設計を行なう。

求められている機能としては、

「貸出」、「返却」、「利用者登録」、「ビデオ追加」、「ビデオ削除」、「プリペイドカード管理」、「金銭管理」、などである。

#### (3) 貸衣裳システムの問題

貸衣裳店の貸衣裳管理システムの設計を行なう。

求められている機能としては、

「貸出」、「返却」、「利用者登録」、「ク

リーニング管理」、「予約」、「金銭管理」などである。

#### 4.2 実践

以下の手順と目的でSPECによる設計を行ない、設計結果や設計過程を評価した。

- ① 図書館システムの問題を新規設計した。
- ② ビデオレンタルシステムの問題を図書館システムの設計結果を用いて設計した。
- ③ ①、②の設計結果から同ドメインのどのシステムにも利用可能なクラスの抽出を行なった。
- ④ ③で定義したクラスを使用して貸衣裳システムの問題を設計した。
- ⑤ ④の結果と比較するため、③で定義したクラスを使用せずに貸衣裳システムの問題を新規設計した。

### 5 考察

以上の実験結果による筆者らの提唱するオブジェクト指向設計技法SPECの他の技法との比較、再利用性に対する定性評価および定量評価結果を以下に示す。

#### 5.1 他の技法との比較

2章で述べた他のオブジェクト指向設計技法との比較結果を以下に示す。

##### ① Boochや大林による設計技法との比較

改造型コンテキストダイアグラムを基にオブジェクトを決定するため、SPECの方がエンティティとシステム間のメッセージやデータの関係によりスムーズにオブジェクトが決定できる。

##### ② F.Manfrediのシステム分析との比較

F.Manfrediによると必ずクラスの設計を行い、クラス間の関係を明確にしていく手法をとっているが、システム全体を定義してからクラス定義に入るSPECの方がよりスムーズな設計ができる。

##### ③ P.Wardらの設計技法との比較

P.WardらはSA／SDによる機能指向の設計でDFDを作成し、更にオブジェクト指向設計をやり直しているが、SPECでは一貫してオブジェクト指向の考え方で要求定義、オブジェクトの抽出を行なっている。

#### ④ S.Shlaerらのオブジェクト指向システム分析との比較

S.Shlaerらは設計については実世界の「もの」をオブジェクトとするしか説明されていないが、SPECでは具体的なオブジェクト・クラスの抽出方法が克明に示されていて、誰が設計しても同じような良い設計結果が得られる。

#### 5.2 再利用性に対する定性評価

新規に設計するときでも、類似システムの設計結果がある場合、設計は容易である。特に、概念図は視覚的表示で直感に訴え再利用しやすい。

また、継承を用いたオブジェクト抽出においては、クラス仕様は作成済みのものをそのまま利用するため記述性の劣化を防止できる。

#### 5.3 再利用性に対する定量評価

##### (1) 概念図の再利用

概念図の作成時間約1／5

図書システム設計時 ···· 10時間

レンタルビデオシステム設計時 2時間

##### (2) オブジェクト・クラス仕様の再利用

(表1に実測データ示す)

##### a) 類似システム利用の場合

記述量に変動無し

##### b) クラス利用の場合

利用しない場合の約1／2

総設計量は約1.5倍(注2)

---

注2) 総設計量の増加は、継承を用いたオブジェクト仕様のコメント、再定義メソッド、クラス仕様書の汎用部分の未削除による。

表1 仕様の再利用調査データ

貸衣裳システム での計測例	ソルタル オブジェクト	組合 オブジェクト	計
(a)汎用システム (元)	66	26	92
貸衣裳システム (b)クラス利用有(元)	40	10	50
(c)クラス+差分 [a+b] (元)	106	36	142
(d)クラス利用無 (元)	72	19	91
実記述量比率 (b/dx100) (%)	56	53	55
総行数比率 (c/dx100) (%)	147	189	160

## 6 今後の課題

### (1) 基本ソフトに対しての検討

本報告の実験ではアプリケーション分野での設計しか行なっていない。OSなどの基本ソフト分野への適用の場合に必要と思われる、性能問題、並列処理の問題に対処するために、それらの問題の設計段階からの検証と設計時の概念図等の文書への描画方法の点でSPECを拡張する必要がある。

### (2) オブジェクト間の関係づけ

類似システムの設計結果や汎用クラスを用いる場合、新たに追加すべき「もの」があればそれを既存のオブジェクト内に吸収するか、新たにオブジェクトを作成するかの基準がない。そのときの、性能、メモリ所用量などへの考慮も重要である。

### (3) オブジェクト仕様／クラス仕様

SPECによる設計後、オブジェクト仕様やクラス仕様からインプリメンテーション作業を行なう。しかし、それらの仕様は日本語で書かれているため、インプリメンテーション時に詳細なデータ構造やメソッドの記述が必要である。インプリメントの時に定義するデータ構造やメソッドを仕様書に示すことができるようすべきである。

### 謝辞

本報告の発表のために御尽力くださった日

科技連ソフトウェア品質管理研究会委員の筑波大学吉澤先生、東京理科大菅野先生に深く感謝致します。また、日科技連ソフトウェア品質管理研究会委員であり、本報告の基となる分科会活動の主査でもあり、本報告の執筆にあたってもご指導くださった日本電気㈱松本正雄氏にも深謝致します。

### 参考文献

- [1] G. Booch: "Object Oriented Design with Applications", Benjamin/Cummings, (1991)
- [2] G. Booch: "Object-Oriented Development", IEEE Trans on SE, SE-12, 2, pp. 211-221 (1986)
- [3] F. Manfredi et.al: "An Object-Oriented Approach to the System Analysis", Lecture Notes in Computer Science Proc. of ESE C'89, Springer-Verlag, pp. 395-410 (1989)
- [4] 大林正晴: "概念による設計法(DMC)による在庫管理システムの記述", 情報処理学会, ソフトウェア工学研究会技術報告, 58-5, pp. 33-40 (1988)
- [5] P. T. Ward: "How to Integrate Object Orientation with Structured Analysis and Design", IEEE Software, pp. 74-82 (March, 1989)
- [6] S. Shlaer, S. J. Mellor (訳: 本位田真一, 山口享) : "オブジェクト指向システム分析", 啓学出版 (1990)
- [7] David Schapiro: "A Continuous Approach to Object-Oriented Software Development", IEEE Comput. Soc. Press, pp. 120-127 (1989)
- [8] 片岡雅憲: ソフトウェアモデリング, 日科技連 (1988)
- [9] P. Coad, E. Yourdon: "Object-Oriented analysis", Youdon Press (1990)
- [10] 中條, 松井, 吉澤, 古賀, 尾崎, 倉山: "オブジェクト抽出手順の確立と再利用性の考察", 第6年度ソフトウェア品質管理研究会、分科会報告書 (非公開), pp. 105-165 (1990)