

5.1.2

オブジェクト指向分析設計

👉 オブジェクトに着目してソフトウェアを設計するのがオブジェクト指向分析設計だ

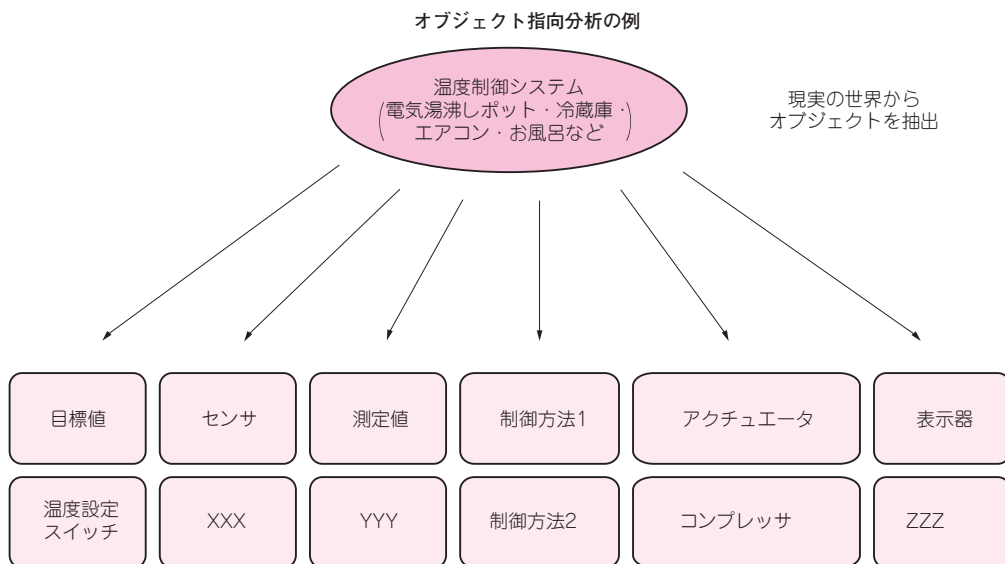
オブジェクト指向分析設計とは、オブジェクト (Object) に着目してシステムを分析し、ソフトウェアを設計する手法である。オブジェクトとは物や動作の対象・目的だが、これら以外に人・機能・データなどもオブジェクトとして扱うのが特徴である。

Key Word 🗝️

オブジェクト, オブジェクト指向(OO), オブジェクト指向分析(OOA), オブジェクト指向設計(OOD), オブジェクト指向分析設計(OOAD), Booch法, OMT法, OOSE法, UML, ラショナル統一プロセス

構造化設計では、プロセスとデータを別々にとらえるので、どちらかに変更が生じると、他方に大きな影響を与え、保守が難しくなる。これに対し、オブジェクト指向設計ではプロセスとデータの両方をもつオブジェクトとしてとらえるので、変更に強くなる。

オブジェクト (Object) とは物や動作の対象・目的だが、オブジェクト指向(OO: Object Oriented)では、これら以外に人・機能・データなどもオブジェクトとして扱う。オブジェクトは、状態、振る舞い、識別性という三つの性質をもっている。そして、現実の世界では一つしか存在しないオブジェクトでもその状態および振る舞いによって、いくつかのオブジェクトに分けることができる。このように、



現実の世界からオブジェクトを抽出して抽象化する作業を、**オブジェクト指向分析(OOA: Object Oriented Analysis)**という。

そして、それぞれのオブジェクトの役割およびオブジェクト間のつながりを後に示すさまざまな図で定義するのが、**オブジェクト指向設計(OOD: Object Oriented Design)**である。オブジェクト指向設計を行うことにより、対象となるモノのもつデータ、機能、動作などを識別して表現でき、品質や柔軟性を向上させ、また多くの技術者による開発体制をとりやすくできる。

オブジェクト指向分析とオブジェクト指向設計をあわせて**オブジェクト指向分析設計(OOAD: Object Oriented Analysis and Design)**という。オブジェクト指向分析設計の手法には、**Booch法**、**OMT法**、**OOSE法**などいろいろな手法が存在したが、現在では、記法は**UML(Unified Modeling Language)**に、開発手法は**ラショナル統一プロセス**に統合されている。

オブジェクト指向分析設計の手法

Booch 法	グラディ・ブーチ(Grady Booch)により提唱された手法である。モデル図においてクラスを独特な記法(破線の雲)で表現するのが特徴的である。この記法は現在では OMT および OOSE の記法と統合され、UML の記法の元となった。また、開発手法も、OMT および OOSE と統合されて、ラショナル統一プロセスとなっている
OMT (Object Modeling Technique) 法 (オブジェクトモデル化技法)	1990 年頃にジェームズ・ランボー、マイケル・ブラハ、ウィリアム・ブレメラニ、フレデリック・エディ、ウィリアム・ローレンセンらにより提唱された手法である。現在では、Booch 法と同様に記法は UML に、開発手法はラショナル統一プロセスに統合されている
OOSE (Object Oriented Software Engineering) 法 (オブジェクト指向ソフトウェア工学)	1992 年にイヴァー・ヤコブソン(Ivar Hjalmar Jacobson)により開発された手法である。ソフトウェアコンポーネントを採用することを提唱し、シーケンス図、コミュニケーション図、ユースケース図、状態遷移図を考案、適用した。現在では、Booch 法、OMT 法と同様に記法は UML に、開発手法はラショナル統一プロセスに統合されている

✓ 要点のチェック

- オブジェクト指向分析設計とは、**オブジェクト(Object)**に着目してシステムを分析し、ソフトウェアを設計する手法である。
- 現実の世界から**オブジェクト**を抽出し抽象化する作業を、**オブジェクト指向分析(OOA)**という。
- 抽出された**オブジェクト**の役割およびオブジェクト間のつながりをさまざまな図により定義するのが、**オブジェクト指向設計(OOD)**である。
- オブジェクト指向分析設計には、**Booch法**、**OMT法**、**OOSE法**などいろいろな手法が存在するが、現在では、記法は**UML**に、開発手法は**ラショナル統一プロセス**に統合されている。

5.1.3

UML

 **オブジェクト指向分析設計の図を統一した記法で表すのがUMLだ**

オブジェクト指向分析設計では、事実上の標準となっているUMLを用いてモデリングを行い、クラス図、オブジェクト図、ユースケース図、アクティビティ図、状態マシン図、シーケンス図などを作成する。

Key Word 

OMG, MUML(Unified Modeling Language), クラス図, オブジェクト図, パッケージ図, コンポジット構造図, コンポーネント図, 配置図, ユースケース図, アクティビティ図, 状態マシン図, シーケンス図, コミュニケーション図, 相互作用概要図, タイミング図, モデリング, 構造図, 振る舞い図, 相互作用図

オブジェクト指向分析設計ではさまざまな手法が提唱されているが、それぞれの手法において作成される図の記法が異なっていたため、1997年にOMG(Object Management Group)によって提案されたのがUML(Unified Modeling Language)という統一記法である。そして現在では、2005年7月に承認されたUML 2.0が標準となっている。

いろいろな手法によって開発されたモデルを、UMLによって表現することができる。統一された表記法なので、開発者だけでなく、ユーザなども理解できる。また、分析、設計工程だけではなく、プログラミング工程でも使用できる。

UMLでは、必要に応じて、クラス図、オブジェクト図、パッケージ図、コンポジット構造図、コンポーネント図、配置図、ユースケース図、アクティビティ図、状態マシン図、シーケンス図、コミュニケーション図、相互作用概要図、タイミング図という13種類の図を作成する。そして、オブジェクト

UMLで作成する13種類の図

クラス図	クラスの構造とクラス間の静的な関係を表現する
オブジェクト図	システムのある時点でのスナップショットを表現する
パッケージ図	モデルの要素をグループ化して表現する
コンポジット構造図	クラスの内部構造を表現する
コンポーネント図	実装したコンポーネントの関係を表現する
配置図	構成するハードウェアの関係を表現する
ユースケース図	システムの役割と外部との関係を表現する
アクティビティ図	システムの振る舞いを表現する
状態マシン図	オブジェクトの状態遷移を表現する
シーケンス図	オブジェクト間の相互作用を時系列に表現する
コミュニケーション図	オブジェクト間の相互作用を関係に着目して表現する
相互作用概要図	相互作用の実行順序を表現する
タイミング図	相互作用と状態遷移に関する時間制約を表現する

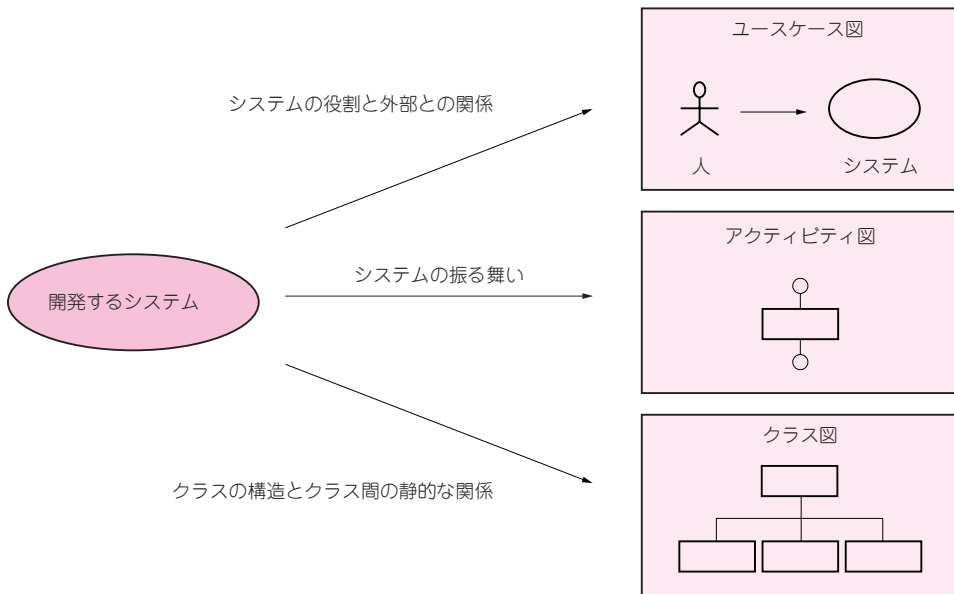
指向分析設計によりこれらの図を作成することを、**モデリング**という。

クラス図、オブジェクト図、パッケージ図、コンポジット構造図、コンポーネント図、配置図は、構造に着目してモデリングされるため、**構造図**と呼ばれる。

ユースケース図、アクティビティ図、状態マシン図、シーケンス図、コミュニケーション図、相互作用概要図、タイミング図は、振る舞いに着目してモデリングされるため、**振る舞い図**と呼ばれる。振る舞い図の中でも、シーケンス図、コミュニケーション図、相互作用概要図、タイミング図は、相互作用に着目してモデリングされるため、**相互作用図**とも呼ばれる。

UMLでは、これら13種類のすべての図を作成する必要はない。よって、極端な例ではクラス図だけでも良いが、クラス図だけでは外部との関係やシステムの振る舞いを表現できないため、そのような場合にはユースケース図やアクティビティ図を作成する。また、とくに組み込みシステムにおいては、メッセージやデータの流れを示すシーケンス図、オブジェクトの状態を示す状態マシン図が重要となる。

オブジェクト指向分析設計によるモデリングの例



✓ 要点のチェック

- 1997年にOMGによって提案されたのが**UML (Unified Modeling Language)**という統一記法である。
- UMLでは、必要に応じて、**クラス図**、**オブジェクト図**、**パッケージ図**、**コンポジット構造図**、**コンポーネント図**、**配置図**、**ユースケース図**、**アクティビティ図**、**状態マシン図**、**シーケンス図**、**コミュニケーション図**、**相互作用概要図**、**タイミング図**という13種類の図を作成し、**モデリング**を行う。
- 組み込みシステムでは、メッセージやデータの流れを示す**シーケンス図**、オブジェクトの状態を示す**状態マシン図**などが重要となる。