

# システム設計論 (改訂版)

工学博士 布 広 永 示  
博士(工学) 今 城 哲 二 共著  
博士(工学) 大 場 みち子  
中 原 俊 政

コ ロ ナ 社

## まえがき

インターネットの発展や WWW (world wide web) の普及に伴い、企業情報システムは、クライアント/サーバシステムから Web システムへと移行しつつある。そして、企業情報システムの中でも、基幹システムは、ユーザのビジネス戦略を左右する大きな要因であり、開発期間の短縮、安定性、高信頼性や高性能が求められる。このような要求に応えるには、システムの特性に応じて、適切なシステムアーキテクチャを選定し、操作性、機能性に優れた設計と構築を行うことが必要である。

本書では、情報社会のインフラストラクチャとして進歩を続ける情報システムについて、集中処理システムから分散処理システム、クライアント/サーバシステム、Web システムへと変化したシステム構成、および情報システムを取り巻く環境や応用分野などについて解説する。さらに、情報システムの開発に必要なシステム分析・設計・構築技術、および情報システムの運用・保守技術について解説する。また、システム分析設計法、トランザクション処理など、情報システムの設計と運用に関連する技術についても説明する。

本書は、情報システムの開発工程の流れと各工程での設計技術を理解し、情報システムの開発に必要な知識や技術の修得を目的としている。

本書の特徴は、情報システムを開発するために必要な開発工程の全体像を示し、各工程の流れや関係をわかりやすく解説していることである。また、情報システムの開発に関連する設計技法、情報システムの運用管理、Web サービス、トランザクション処理などに関する内容も解説している。

本書は、10 の章から構成されている。

1 章では、ハードウェアの性能向上や小型化・低価格化に対する情報システムの処理形態の変遷、および情報システムの開発技法の概要などについて解説

する。

2章では、情報システムが集中処理システムから分散処理システム、そしてクライアント/サーバシステムへと変化する過程とクライアント/サーバシステムを Web 化するための技術について解説する。

3章では、システム設計・構築の開発モデル、工程の流れ、テスト計画・設計、プロジェクト管理、CASE ツール、共通フレームについて解説する。

4章では、従来のホスト系の開発にはなかったオープン系の開発サイクルの一つであるインフラ設計について解説する。

5章では、システム設計の分析設計技法の代表的な二つの設計技法である構造化分析設計技法とオブジェクト指向分析設計技法について図式表現を中心に解説する。

6章では、Web 技術を利用したアプリケーションの設計・構築の流れやユーザインタフェース設計について解説する。

7章では、データベースを管理するデータベースシステムの構成やデータベース設計・構築の流れについて解説する。

8章では、システムの構築とテスト、およびシステムの運用設計と管理について記述する。また、運用管理に関するフレームワークとして標準化されている ITIL についても解説する。

9章では、汎用的なデータ記述言語 XML の特徴やその使い方と XML を応用した Web サービスのしかけや事例について解説する。

10章では、銀行のオンラインや航空機の座席予約、企業の生産管理、受発注管理、物流管理などの巨大システムで、膨大なデータと多数の利用者を相手にするトランザクション処理の特性や実行に必要な機能などについて解説する。

最後になりましたが、本書の構成は、東京国際大学 今城先生からのご助言を参考にして進めてきました。今城先生には、1章「情報システムとその設計思想の変遷」、5章「システム分析設計技法」のご執筆をしていただきました

が、残念ながら、本書の完成を目にすることなく出版の2か月前に他界されました。

しかし、本書の校正作業を、病床で最後の最後まで進められたことを考えると、システム設計に対する今城先生の思いを本書に残されたのではないかと思います。

本書の共同執筆者である、バブ日立ソフト株式会社 中原，株式会社日立製作所 大場，東京情報大学 布広は、今城先生が日立製作所に在籍されていたときから、技術面、精神面に渡ってご指導を受けた仲であり、いろいろな場面で的確な指示をいただきました。

本書を今城先生に捧げ、ご冥福をお祈りいたします。

2007年8月

布広 永示

## 改訂版の出版にあたって

近年、システム開発のスタイルは、ビジネスの変化に柔軟に対応し、効率良く開発するために変化してきている。システム開発モデルでは、従来のウォーターフォールモデル中心からアジャイルなソフトウェア開発の一部適用、Webサービスを使用した開発では、設計思想スタイルが多様化し、SOAP、WSDL、UDDI、XMLだけでなく、Webシステムと相性が良いRESTやJSONが普及してきている。また、クラウド環境が提供するサービスをWebサービスとして利用できるようになった。そこで改訂版では、これらに対応して記述内容の見直しを行った。また、Webサービスについては事例を追加し、本書に関係する規格として、PMBOKガイド第5版、共通フレーム2013、システム運用設計のフレームワークITIL V3の発行に対応した最新の規格の内容に変更した。

2017年7月

布広 永示

# 目 次

## 1. 情報システムとその設計思想の変遷

1.1 情報システムの処理形態とその変遷 .....	1
1.1.1 企業情報システムの処理形態 .....	1
1.1.2 バッチシステム .....	3
1.1.3 オンラインシステム：集中処理から分散処理へ .....	3
1.1.4 OLTP システムとデータベース管理システム .....	4
1.1.5 オンラインシステムのパターン：2階層と3階層 .....	5
1.2 ハードウェアの急速な進歩 .....	6
1.3 恒常的なソフトウェア危機と解決への努力 .....	8
1.3.1 コンピュータ向きのアセンブラ ⇒ 問題分野向き的高级言語 .....	8
1.3.2 量の問題（開発規模の増大）⇒ 構造化技法とソフトウェア工学 .....	8
1.3.3 質の問題（信頼性、使い勝手、多重外注） .....	9
1.4 情報システムの設計思想 .....	10
1.4.1 システム開発モデル .....	10
1.4.2 モデル化技法（システム分析のための世界観） .....	12
演 習 問 題 .....	16

## 2. クライアント/サーバシステム

2.1 情報システムの変革 .....	17
2.1.1 集中処理システム .....	17
2.1.2 分散処理システム .....	19

2.2 クライアント/サーバシステムの登場	23
2.2.1 クライアント/サーバシステムとは	23
2.2.2 クライアント/サーバシステムの構成	24
2.3 Web 3 階層型クライアント/サーバシステム	29
2.3.1 Web システムを実現するための構成要素	29
2.3.2 Web 技術を利用した 3 階層型 C/S システム	31
2.3.3 Web システムに求められる要件	33
演習問題	34

### 3. システム設計・構築

3.1 システム開発モデル	35
3.1.1 ウォータフォールモデル	35
3.1.2 プロトタイピングモデル	36
3.1.3 スパイラルモデル	37
3.1.4 アジャイルなソフトウェア開発	38
3.2 システム設計・構築の流れ	39
3.2.1 システム設計の種類と工程の流れ	39
3.2.2 システム化計画	40
3.2.3 要求分析	41
3.2.4 プロジェクト計画	42
3.2.5 見積り	42
3.2.6 インフラ設計	44
3.2.7 運用設計	45
3.2.8 アプリケーション設計・構築	45
3.2.9 データベース設計・構築	47
3.2.10 テスト	47
3.2.11 運用管理	48
3.3 テスト計画・設計	48
3.4 プロジェクト管理	50
3.5 CASE ツール	53

3.5.1 CASE ツールとは .....	53
3.5.2 リポジトリ .....	54
3.5.3 リエンジニアリング .....	54
3.6 共通フレーム .....	55
演習問題 .....	56

## 4. インフラ設計

4.1 インフラ設計とは .....	57
4.2 インフラ設計の流れ .....	60
4.2.1 業務要件の分析 .....	60
4.2.2 業務のパターン化 .....	61
4.2.3 インフラパターンの決定 .....	63
4.2.4 インフラの選定 .....	63
演習問題 .....	65

## 5. システム分析設計技法

5.1 構造化分析設計技法 .....	66
5.1.1 記述例で使用するアプリケーションの概要 .....	66
5.1.2 構造化分析 .....	67
5.1.3 構造化設計 .....	72
5.2 オブジェクト指向分析設計技法 .....	76
5.2.1 UML .....	76
5.2.2 用例図 (ユースケース図) .....	78
5.2.3 クラス図 .....	79
5.2.4 MDA .....	80
演習問題 .....	81

## 6. アプリケーション設計・構築

6.1 アプリケーション設計・構築の流れ	82
6.1.1 外部設計	83
6.1.2 内部設計	86
6.1.3 プログラム設計	88
6.1.4 プログラミング	90
6.1.5 テスト	90
6.2 ユーザインタフェース設計	94
6.2.1 GUIの操作性	95
6.2.2 GUI設計の流れ	96
6.3 Web技術を利用したアプリケーション設計	98
6.3.1 Webアプリケーションとは	98
6.3.2 Webアプリケーションの基本構造	99
6.3.3 Webアプリケーションの機能設計	99
6.3.4 サーバ側で動作するアプリケーションの実行方法	101
6.3.5 性能設計	103
6.3.6 運用に関わる設計方針	104
演習問題	105

## 7. データベース設計・構築

7.1 データベースシステムとは	106
7.1.1 ファイル管理の問題点	106
7.1.2 データベースシステム	107
7.2 データベース管理システム	108
7.2.1 DBMSの役割	108
7.2.2 DBMSの機能	109
7.3 データ独立性と3層スキーマ構造	110

7.4 データモデル	113
7.4.1 概念データモデル	114
7.4.2 論理データモデル	114
7.4.3 物理データモデル	115
7.5 データベース設計・構築の流れ	115
7.5.1 概念設計	116
7.5.2 論理設計	120
7.5.3 物理設計	126
演習問題	127

## 8. システム運用設計・運用管理

8.1 システムの構築・テストと運用管理の流れ	128
8.1.1 システムの構築	128
8.1.2 システムテスト	129
8.1.3 運用テスト	131
8.1.4 運用管理	131
8.2 システム運用設計のフレームワーク：ITIL	133
8.2.1 ITIL とは	133
8.2.2 運用設計	135
8.3 セキュリティポリシーの構築・運用	137
8.3.1 情報セキュリティポリシーとは	137
8.3.2 セキュリティポリシーの策定	138
8.3.3 セキュリティシステムの導入	139
8.3.4 セキュリティシステムの運用	140
8.3.5 情報セキュリティマネジメントシステム	141
演習問題	141

## 9. データ記述・表現と Web サービス

9.1 データ記述と表現：XML と HTML .....	142
9.1.1 データ記述言語：XML .....	142
9.1.2 データの構造定義：DTD と XML Schema .....	145
9.1.3 Web 上でのデータ表現：HTML .....	147
9.1.4 XML の応用動向 .....	148
9.2 Web サービス .....	150
9.2.1 Web サービスの定義と出現の背景 .....	150
9.2.2 Web サービスを支える技術とメカニズム .....	152
9.2.3 クラウド環境と Web API .....	157
9.2.4 WebAPI を利用した Web サービスの事例 .....	160
9.3 SOA .....	162
9.3.1 SOA のアプローチ .....	162
9.3.2 SOA でのサービスの考え方 .....	163
9.3.3 SOA を実現するための技術 .....	165
9.3.4 マイクロサービス .....	166
演習問題 .....	166

## 10. トランザクション処理

10.1 トランザクション処理とは .....	167
10.2 ACID 特性 .....	168
10.3 トランザクション処理モニタ .....	169
10.3.1 コミットとロールバック .....	170
10.3.2 キュー管理 .....	170
10.3.3 トランザクション処理の実行例 .....	171
10.4 2相コミットメント .....	173
10.5 障害回復 .....	174

10.6 排 他 制 御 .....	176
10.6.1 排他制御の種類 .....	176
10.6.2 排他制御の必要性 .....	177
10.6.3 デッドロック .....	178
10.7 システムの高性能・高信頼化技術 .....	181
10.7.1 システムの高性能化技術 .....	181
10.7.2 システムの高信頼化技術 .....	185
10.7.3 分散オブジェクト技術 .....	193
演 習 問 題 .....	194
<b>引用・参考文献 .....</b>	<b>195</b>
<b>演習問題解答 .....</b>	<b>199</b>
<b>索 引 .....</b>	<b>204</b>

# 情報システムとその設計思想の変遷

企業情報システムの処理形態としてはバッチ処理とオンライン処理があり、長い間これらはコンピュータ室に設置されたホストコンピュータを中心とした集中処理で行われていた。1990年代のワークステーションやパーソナルコンピュータ（PC）の普及に対応し、複数のコンピュータによるネットワークを介した分散処理が主流となった。2000年以降に開発されたオンラインシステムの多くは、インターネットのWebブラウザが活用されている。

約60年のコンピュータの歴史の中で、ハードウェアは約2年ごとに倍の性能になるなど急激な性能向上と小型化・低価格化が今でも継続しているが、それと比ベソフトウェアの進歩は遅々としており、品質や生産性などの多くの問題がある。これらを解決するために、システム開発を分析・設計・構築（プログラミング）・テストの工程に分けて実施するというウォーターフォールモデルが定着し、工程ごとに各種の分析技法、設計技法、プログラミング技法が提案され、実用化されてきている。

## 1.1 情報システムの処理形態とその変遷

### 1.1.1 企業情報システムの処理形態

民間の企業や官公庁・地方自治体などの情報システム（以下、企業情報システムと略す）は、図1.1のようにネットワークでサーバと端末が結ばれ、データベースを中核として、オンライン処理形態とバッチ処理形態の業務が共存している。

人が介入する必要がある対話型の処理はオンライン処理で行われ、完全に自動化が可能な非対話型の処理はバッチ処理で行われる。オンライン処理とは、サーバと端末をネットワークで結び、端末からデータが発生するたびに即座に

## 2 1. 情報システムとその設計思想の変遷

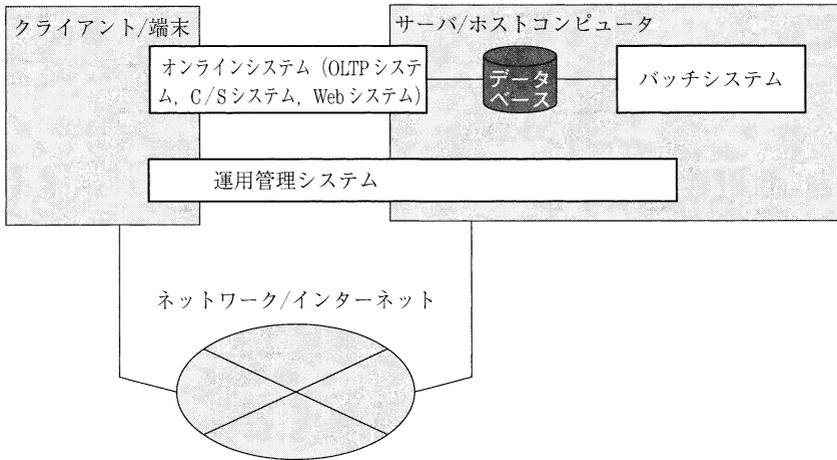


図 1.1 企業情報システムの標準的な処理形態

処理する方式である。オンライン処理には、以前からある OLTP (online transaction process : オンライントランザクション処理) システムと、1990 年代以降のおもに PC がクライアントの C/S (client/server : クライアント/サーバ) システム、さらに 2000 年以降に急速に普及したインターネットの Web ブラウザを使った C/S システム (特に Web システムと呼ぶ) の三つの形態が存在する。一方、バッチ処理とは一定期間あるいは一定量のデータを集め、一括で処理する方式である。複数の手順からなる定型の処理を登録しておき、自動的に連続処理を行う。

企業情報システムがこのように複雑になっているのは、いろいろな時期に開発された多くの業務システムが共存しているからであり、古いものでは 30 年以上前に開発され、その後更新されながら使用され続けているものまで存在する。

本書の中心テーマは、C/S システムの設計方法である。2 章で C/S システムについて説明し、3 章以降でシステムを設計するときの個々の話題について議論する。複雑な企業情報システム全体を円滑に運用するには、運用管理システムも重要である。運用管理システムについては、特に 8 章で詳しく説明する。

### 1.1.2 バッチシステム

コンピュータが使われ始めた1950年代から1960年代は、コンピュータは鍵のかかったコンピュータ室に設置され、中央のコンピュータに入出力機器（カードリーダー、紙テープリーダー/パンチ、プリンタ、磁気テープ装置など）と補助記憶装置（ディスク）が接続されていた。業務はすべてバッチ処理で行われた。バッチ処理で行われるシステムをバッチシステムという。代表的な業務は、給与計算システムである。このシステムは、カードあるいは紙テープに記録された従業員の1か月の出勤日数や残業時間などの勤務データを入力し、ディスクのファイル上にある個人個人の基本給などを参照して、給与や天引きされる税金などを計算し、一人ひとりに配る給与票をプリンタで印刷する。今でも多くの企業の給与計算システムは、バッチ処理で実施されている。

### 1.1.3 オンラインシステム：集中処理から分散処理へ

1960年前後にコンピュータを用いて飛行機や列車の座席予約をオンラインで行うシステムが始まった。その後、銀行の預金システムなどがオンライン化され、1970年代から個々の企業内のシステムも順次オンライン化され、1980年代はそれが当たり前になった。

オンラインシステムは、1960年代にコンピュータの集中処理として始まった。集中処理時代のコンピュータをホストコンピュータあるいはメインフレームと呼ぶ。それにネットワークの端にある端末が複数台つながっていた。主役はホストコンピュータという考え方である。集中処理時代は長く続いたが、1990年代のワークステーションやPCの普及に対応し、一つの業務を複数のコンピュータで分業する分散処理が普及した。それが、C/Sシステムである。C/Sシステムでは主役は端末側に移り、クライアントと呼ぶ。クライアントの要求に応じてサービスを行うのがサーバである。通常、クライアント側はPC、サーバ側はワークステーション、あるいはUNIXかWindows Server搭載のPCである。

集中処理の時代と分散処理の時代の大きな違いは、前者の時代では端末のコ

コンピュータパワーが貧弱だったので、端末側は CUI (character user interface) による画面の入出力しか行えなかった。一方、後者の時代は端末側 (クライアント, おもに PC) に十分なコンピュータパワーがあり, GUI (graphical user interface) による画面処理や業務処理が行えるようになったことである。さらに, 2000 年代にはインターネット技術の普及に対応し, オンラインシステムの端末として Web ブラウザが使用されるようになった。これを Web システムという。Web システムは C/S システムの一形態であるが, 端末側では Web ブラウザが使用されるので, 端末側では画面処理や業務処理は行わない。この方式は, 集中処理時代の端末と似ている。

#### 1.1.4 OLTP システムとデータベース管理システム

オンラインシステムの中で特に「トランザクション制御」, 「性能」, 「信頼性」が求められるシステムには, トランザクション処理モニタ<sup>†</sup> (TP モニタ: transaction processing monitor, オンライン管理プログラムともいう) を用いる。TP モニタを用いたものを, 特に OLTP システムという。TP モニタのおもな機能は, OLTP を管理し, 障害の回復, トランザクションのスケジューリング管理, システム資源の管理などである。集中処理時代のオンラインシステムでは, ほとんどの場合 TP モニタを使用していた。C/S システムや Web システムでも, 特に性能や信頼性を追求するシステムでは TP モニタを用いる。

バッチシステムでは, データはファイルの中に保管していた。バッチで複数のプログラムを同時に実行するときには同じファイルを同時に更新しないように運用を注意しなければならなかった。また, 例えば従業員コードなど同一データが複数のファイルに存在することも多くあり, 従業員コードの改変時には, 関連するすべてのファイルを更新する必要があった。このようなファイルシステムの欠点を解決するために, 1970 年代に登場したのがデータベースで

---

<sup>†</sup> 10.3 節参照。

ある。特にオンラインシステムでは、データを集中管理するとともに、複数端末からの同一データの更新を矛盾が生じないように制御する必要があるので、データベースを使うのが当たり前となった。データベースを管理するプログラムを、データベース管理システム<sup>†</sup> (DBMS : database management system) と呼び、OLTP システムでは TP モニタと連動して動作する。

### 1.1.5 オンラインシステムのパターン：2階層と3階層

C/S システムのオンライン処理の業務アプリケーションを設計するとき、画面処理 (ユーザインタフェース処理) と業務 (application : アプリケーション, AP) 処理を分離して設計するか、一緒にして設計するかの二つの方式がある。

分離するときは、クライアントで画面処理を行い、サーバで業務処理を行う。さらにサーバ側は、業務処理を行うアプリケーションサーバ (AP サーバ : application server) とデータベース管理 (DB 管理 : database management) を行うデータベースサーバ (DB サーバ : database server) に分けるのが普通である。このような階層に分けて設計する方式を、3階層システムと呼ぶ。画面処理と業務処理をクライアントで行い、サーバ側でも業務処理の一部と DB 管理を行う方式を、2階層システムと呼ぶ。

端末側のユーザインタフェースのニーズの変化や端末自体の技術変化にともない画面処理の変更頻度は多いが、それと比べ業務処理の変更はそれほど多くない。3階層の考えの基本は、これらを一緒にせず、変わるものと変わらないものを分離しておくというものである。クライアントが PC の場合と Web ブラウザの場合のそれぞれに2階層と3階層があるので、合計四つのパターンとなる。これを表 1.1 に示す。表 1.1 にはホストコンピュータのパターンも加えてある。詳しくは2章で説明する。

---

<sup>†</sup> 7.2 節参照。

# 索引

<b>【あ】</b>	ガントチャート	52	コミット	170
アジャイル開発手法	38	<b>【き】</b>	<b>【さ】</b>	
アトリビュート	69, 117	機能階層図	サーバ	24
アプリケーションサーバ	5	機能テスト	サービス	162
<b>【い】</b>	キュー管理	170	サービスオペレーション	
一貫性	168	協調モデル		134
インスペクション	93	共通フレーム	サービスストラテジ	134
インフラ	44	業務機能コンポーネント	サービスデザイン	134
インフラ設計	44		サービストランジション	
<b>【う】</b>				134
ウォークスルー	93	<b>【く】</b>	サブレット	102
ウォーターフォールモデル	10, 35	クライアント	3階層型C/Sシステム	26
運用管理	48	クライアントアプリ	3階層C/S設計技法	83
運用設計	45	セッション	3層スキーマ構造	110
運用テスト	48	クラウド環境		
<b>【え】</b>	クラウドコンピューティング	157	<b>【し】</b>	
エンティティ	69, 117	クラス図	システム管理	20
<b>【お】</b>	クラスタリング	183, 189	システム機能コンポーネント	164
オブジェクト指向モデル	15	クロック周波数	システムテスト	48
オブジェクト図	77	<b>【け】</b>	持続性	168
オンプレミス	157	継続のサービス改善	集中処理システム	17
オンラインシステム	3, 18	系列図	状態機械図	78
<b>【か】</b>	結合テスト	47	状態遷移モデル	15
カーディナリティ	118	限界値分析法	情報セキュリティマネジ メントシステム	141
概念スキーマ	111	原子性	168	
概念設計	47, 115	<b>【こ】</b>	<b>【す】</b>	
概念データモデル	114	交信図	垂直分散システム	20
外部スキーマ	111	合成構造図	水平分散システム	20
外部設計	45	構造化設計	スケルトン	194
活動図	78	構造化定理	スタブ	92, 194
		構造化プログラミング	ステップ法	42
		構造化分析設計技法	ステートフル	155
		構造図	ステートレス	155
			スパイラルモデル	12, 37

スループット	182	ドライバ	93	物理的データ独立性	112
<b>【せ】</b>		トランザクション処理	167	部品図	77
制御フローモデル	14	トランザクション処理		ブラックボックステスト	94
性能テスト	130	モニタ	4, 169	フルブルーフ	189
<b>【そ】</b>		<b>【な】</b>		プレゼンテーション層	26
相互作用概観図	77	内部スキーマ	111	プログラミング	46
相互作用図	77	内部設計	46	プログラム設計	46
操作性テスト	130	<b>【に】</b>		プロジェクト管理	50
ソフトウェア管理	21	2階層型C/Sシステム	25	プロセス分析	70
ソフトウェアライフ		2相コミットメント	173	プロトコル	194
サイクルプロセス	55	<b>【ね】</b>		プロトタイピングモデル	11, 36
<b>【た】</b>		ネットワーク管理	21	分散オブジェクト技術	193
ターンアラウンドタイム	182	<b>【は】</b>		分散処理システム	19
第1正規化	121	排他制御	176	分離性	168
第2正規化	121	配置図	77	<b>【へ】</b>	
第3正規化	121	バッチシステム	3, 18	平均故障時間間隔	186
タイミング図	78	判定条件網羅	91	平均修理時間	186
単体テスト	46	<b>【ひ】</b>		並列処理	22, 183
<b>【て】</b>		ビジネスコンポーネント	164	<b>【ほ】</b>	
テスト工程品質管理図	53	ビジネスプロセスの記述	165	包装図	77
データ中心アプローチ	113	ビッグバンテスト	93	保全対策テスト	131
データの正規化	121	<b>【ふ】</b>		ホットスタンバイ	189
データフロー図	68	ファンクション層	26	ボトムアップテスト	93
データフロー分析	68	ファンクションポイント法	43	ボトムアップ見積り法	44
データフローモデル	13	フェイルオーバー	190	ホワイトボックステスト	91
データ分析	69	フェイルセーフ	189	<b>【ま】</b>	
データベース管理システム	5, 108	フェイルソフト	189	マイクロサービス	166
データベースサーバ	5	フォールトトレランス	189	マークアップ言語	143
データベースシステム	107	フォールトトレラント		マルチプロセスシステム	189
データベース層	26	システム	188	<b>【み】</b>	
データモデル	113	フォールトトレラント設計	189	ミラーリング	191
デッドロック	178	技術	189	<b>【め】</b>	
デュアルシステム	188	負荷テスト	131	命令網羅	91
デュプレックスシステム	187	負荷分散	22, 184	<b>【も】</b>	
<b>【と】</b>		複数条件網羅	91	モジュール階層図	73
同値分割法	94	物理設計	47, 116	モデル化	13
独立性	110	物理データモデル	115		
トップダウンテスト	92				

	<b>【ゆ】</b>	粒度	163	レスポンスタイム	182
ユースケースモデル	78	リレーショナルデータ		<b>【ろ】</b>	
	<b>【よ】</b>	ベース管理システム	108	ロールバック	170, 174
用例図	78	リレーショナルデータ		ロールフォワード	174
	<b>【り】</b>	モデル	113	論理設計	47, 116
リエンジニアリング	54	リレーションシップ	70, 117	論理データモデル	114
リストラクチャリング	54	<b>【る】</b>		論理的データ独立性	112
リバースエンジニアリング	54	類似見積り法	42	<b>【わ】</b>	
リポジトリ	54	<b>【れ】</b>		ワークパッケージ	52
		レグレッションテスト	131		
◆ ◆					
	<b>【A】</b>	HTML	29, 147	SOA	162, 165
ACID 特性	168	<b>【I】</b>		SOAP	152
AP サーバ	5, 24, 33	IaaS	157	SPEC	182
	<b>【B】</b>	ITIL	133	SQL	114
BPMN	165	<b>【J】</b>		<b>【T】</b>	
	<b>【C】</b>	JSON	156	TCO	132
CASE ツール	53	<b>【M】</b>		TPC	183
CGI	101	MDA	80	TP モニタ	4, 169
COCOMO 法	43	MIPS	181	<b>【U】</b>	
CORBA	193	<b>【O】</b>		UDDI	153
	<b>【D】</b>	OLTP システム	4	UML	76
DB サーバ	5, 20, 24, 33	OMG	193	<b>【W】</b>	
DBMS	5, 108	<b>【P】</b>		WBS	44
DDL	115	PaaS	157	Web アプリケーション	31, 98
DML	115	PDCA	140	Web サーバ	30
DTD	144	PMBOK	50	Web サービス	150
	<b>【E】</b>	<b>【R】</b>		Web ブラウザ	30
ER 図	69, 117	RAID	191	WebAPI	158
ER モデル	117	RASIS	185	WSDL	152
ESB	163	REST	155	<b>【X】</b>	
	<b>【F】</b>	<b>【S】</b>		XML	142
FLOPS	181	SaaS	157		
	<b>【H】</b>	SGML	143		
HIPO 図	72	SLA	136		

— 著者略歴 —

**布広 永示** (ぬのひろ えいじ)

- 1980年 日本大学生産工学部数理工学科卒業
- 1982年 日本大学大学院生産工学研究科博士前期課程修了 (数理工学専攻)
- 1985年 日本大学大学院生産工学研究科博士後期課程単位取得満期退学 (数理工学専攻)
- 1985年 株式会社日立製作所勤務
- 1987年 工学博士 (日本大学)
- 2002年 東京情報大学助教授
- 2007年 東京情報大学教授
- 現在に至る

**今城 哲二** (いまじょう てつじ)

- 1969年 国際基督教大学教養学部自然科学科数学専攻卒業
- 1969年 株式会社日立製作所勤務
- 2002年 奈良先端科学技術大学院大学情報科学研究科博士後期課程修了 (情報科学専攻) 博士 (工学)
- 2004年 東京国際大学教授
- 2007年 逝去

**大場 みち子** (おおば みちこ)

- 1982年 日本女子大学家政学部家政理学科 (物理学専攻) 卒業
- 1982年 株式会社日立製作所勤務
- 2001年 大阪大学大学院工学研究科博士後期課程修了 (情報システム工学専攻) 博士 (工学)
- 2007年 東京工科大学・駒澤大学非常勤講師
- 2010年 公立はこだて未来大学教授
- 現在に至る

**中原 俊政** (なかはら としまさ)

- 1973年 鳥取大学工学部電子工学科卒業
- 1973年 株式会社日立製作所勤務
- 2005年
- ～11年 パブ日立ソフト株式会社勤務
- 2007年 東京情報大学非常勤講師
- 2010年 技術士 (情報工学)

## システム設計論 (改訂版)

System Design (Revised Edition)

© Nunohiro, Imajo, Oba, Nakahara 2007

2007年10月25日 初版第1刷発行

2017年9月25日 初版第6刷発行 (改訂版)

検印省略

著者	布 広 永 示 今 城 哲 二 大 場 み ち 子 中 原 俊 政
発行者	株式会社 コロナ社 代表者 牛来真也
印刷所	壮光舎印刷株式会社
製本所	株式会社 グリーン

112-0011 東京都文京区千石4-46-10

発行所 株式会社 コロナ社  
CORONA PUBLISHING CO., LTD.

Tokyo Japan

振替00140-8-14844・電話(03)3941-3131(代)

ホームページ <http://www.coronasha.co.jp>

ISBN 978-4-339-02878-2 C3055 Printed in Japan

(柏原)



 < 出版者著作権管理機構 委託出版物 >

本書の無断複製は著作権法上での例外を除き禁じられています。複製される場合は、そのつど事前に、出版者著作権管理機構（電話 03-3513-6969, FAX 03-3513-6979, e-mail: info@jcopy.or.jp）の許諾を得てください。

本書のコピー、スキャン、デジタル化等の無断複製・転載は著作権法上での例外を除き禁じられています。購入者以外の第三者による本書の電子データ化及び電子書籍化は、いかなる場合も認めていません。落丁・乱丁はお取替えいたします。