

マルチエージェントシリーズ B-2

# マルチエージェントによる 自律ソフトウェア設計・開発

大須賀昭彦  
田原 康之  
中川 博之  
川村 隆浩

共著

コロナ社

マルチエージェントシリーズ  
編集委員会

編集委員長

寺野 隆雄 (東京工業大学, 工学博士)

編集委員

和泉 潔 (東京大学, 博士 (学術))

伊藤 孝行 (名古屋工業大学, 博士 (工学))

大須賀 昭彦 (電気通信大学, 工学博士)

川村 秀憲 (北海道大学, 博士 (工学))

倉橋 節也 (筑波大学, 博士 (システムズ・マネジメント))

栗原 聡 (電気通信大学, 工学博士)

平山 勝敏 (神戸大学, 博士 (工学))

松原 繁夫 (京都大学, 博士 (情報学))

## 刊行のこぼ

21世紀に入り、人間の活動の世界規模での展開と情報通信をはじめとする技術の急速な発展普及に伴って、世界規模で人々の意識や行動の変化が、既存の社会や制度に追いつかない現象が頻発している。例えば、世界で頻発する文化的な摩擦やテロなどの事象、鳥インフルエンザなどの感染症の流行、SNSなどのネット上での人々のコミュニティ行動の理解、電子商取引の発展、金融市場の不安定性などはその例である。これらに共通する問題はつぎの3点である。(1) 対象が本質的に変動し続ける性質を持ち、物理現象のような第一原理が存在しないこと、(2) 対象となる現象を分析するという従来の自然科学的接近法に加えて、対象をデザインするという新しい工学的接近法が必要なこと、(3) 当事者や関係者を含む複雑な意思決定という側面を持ち、対象問題を定式化することが非常に困難であること。

このような複雑な現象の分析、設計においては、従来とは異なり、対象となるシステムが所与のものとは仮定することはできない。システム全体を表す法則が、システムを構成する要素の相互作用から創発しうるからである。われわれはこのような社会的・システムの課題について「マルチエージェント」の概念を用いることで新しい方法論が構築できると信じている。マルチエージェントとは、エージェントと呼ぶ内部状態と意思決定・問題解決能力、ならびに通信機能を備えた複数の主体によるボトムアップなモデル化を試みる。そしてこのインタラクションに基づく創発的な現象やシナリオを分析しようとする。

近年、マルチエージェントが注目されるようになった背景には、コンピュータそのものの急速な発展、オブジェクト指向などのソフトウェア開発手法の進歩、進化や学習を扱う人工知能技術の発展、分岐・相転移やカオス、自己組織化などを扱う非線形科学や複雑系科学の発展が挙げられる。そして、このよう

## ii 刊 行 の こ と ば

な理論や手法を適用するためには、コンピュータによるシミュレーションによる理解が必要になる。

この古くて新しいシミュレーションの考え方は、対象をモデル化し数理的に扱う演繹的な方法、もしくは、データや事例分析を用いる機能的な方法を補完する第3の科学的方法であり、複雑な事象に対するわれわれの直観の能力を高める性質を持つ。マルチエージェントの考え方は、したがって、計算機科学をはじめとする理科系の学生にとっても、経済学、社会学などを学ぶ文科系の学生にとっても、研究の道具として、また、複雑な社会現象を知るための教養として、今後、必須のものになると考えられる。

本シリーズのねらいは、このような複雑システムの分析、設計に伴う困難を克服する手段としてのマルチエージェント理論や技術について体系付けて学ぶ機会を提供することである。本シリーズでは、全体を通じて、新しい学際的な方法としてのマルチエージェントの考え方を紹介し、それに基づいたマップを示す。本シリーズの大きな特長は、各巻において、ほかの巻の内容との関連性を明示するとともに、Webサイトを積極的に利用して、スライドやプログラムソース、シミュレーション実行例などの副教材を豊富に提供することである。このような試みはわが国においても、世界的にもはじめてである。この新たな学際領域に、みなさんを招待したいと考える。

2017年5月

編集委員長 寺野 隆雄

# ま え が き

近年のインターネットおよび Web の大規模かつ急速な発展は社会を大きく変えつつある。ハードウェア・ソフトウェア双方の技術の進歩により、さまざまなデバイスをネットに接続して通信させることが可能となり、いわゆるモノのインターネット (Internet of Things : IoT) と呼ばれる環境が急速に普及している。またビッグデータ処理技術、人工知能技術、およびクラウドコンピューティング技術などの急速な進展により、利便性の高いサービスを迅速に提供し、ユーザーニーズの変化に応じて柔軟に対応することが求められるようになってきている。しかし Web 環境で稼働するソフトウェアの開発、運用においては、このような大規模かつ急速な変化に的確に対応することはますます困難になりつつある。

本書は、このような課題の解決のために、マルチエージェント技術を活用したソフトウェア、特に Web 関連のソフトウェアに関する要素技術、開発手法、および応用について解説したものである。エージェントは従来のソフトウェアに比べ、さまざまな先進的な特徴により、近年の大規模複雑なシステムの実現におおいに貢献している。本書ではそれらの特徴のうち、特に「自律性」に着目している。本書においてエージェントの自律性とは、自ら達成を目指す目的を持ち、自ら環境とインタラクションを行うことにより目的を達成する能力のこととする。このような特徴により、急速に変化する環境下でも、可能な限り不具合を発生させたり停止したりすることなく、継続的に機能し続けることが可能となる。Web はまさにそのような変化の激しい環境であり、自律エージェントの活用がおおいに有望である。

1 章では、現在のマルチエージェントシステム技術の理解を深めるうえで有用な、自律ソフトウェア開発手法の歴史的背景について解説する。2 章では、3 章以降で紹介する自律ソフトウェアのアプリケーションの設計に必要な要素技

術として、数理論理学、プランニング、ソフトウェア工学、機械学習、自然言語処理、および複雑ネットワークなどについて概説する。3章では、ソフトウェアへの要求を適切に抽出・定義・管理する手法として、マルチエージェントシステムなどの自律ソフトウェア構築にも適したアプローチであるゴール指向要求工学について概説する。4章では、自律ソフトウェアの一種として期待が高まっている、自己適応システム、すなわち環境の変化を検知し、必要に応じて自分自身を再構成することによって環境の変化に適応しようとするシステムを実現する技術について概説する。5章では、Webを発明した Tim B. Lee 氏によって 1998 年に新しく提唱されたセマンティック Web 上の情報に基づいてさまざまなサービスを行う、セマンティック Web エージェントについて概説する。6章では、Web 上の急速に変化する膨大な情報を有効活用するために、自律ソフトウェア技術を応用した Web アプリケーションについていくつかの事例を紹介する。

本書は、つぎの 4 名で執筆した。

大須賀昭彦：1, 6 章

田原康之：2 章

中川博之：3, 4 章

川村隆浩：5 章

2 章は 3~6 章の内容を理解するために必要な基礎知識を概説しているので、3~6 章の前に読むのが適切と考えられるが、それ以外については比較的独立しているため、必要などころから読んでいけばよい。なお、本書に掲載されているスライドは、本書の書籍詳細ページ<sup>†</sup>でダウンロードが可能なので、ぜひ活用いただきたい。

本書は多くの方々のご協力のもとで刊行に至ったものと考えている。まず、本書執筆の機会を与えていただいた、マルチエージェントシリーズ編集委員会に深く感謝する。つぎに、日頃ご指導いただいている、東京大学大学院情報理工学系研究科/国立情報学研究所 本位田真一教授、早稲田大学理工学術院基幹

---

<sup>†</sup> <http://www.coronasha.co.jp/np/isbn/9784339028188/>

理工学部 深澤良彰教授，および大阪大学大学院情報科学研究科 土屋達弘教授に深く感謝する。また，日頃熱心にご議論いただいている，電気通信大学情報理工学研究科 清雄一助教に深く感謝する。そして，本書執筆に際して多大なご協力を頂いた，株式会社コロナ社に深く感謝する。最後に，日頃執筆者一同とともに研究成果を生み出してきた，電気通信大学 大須賀・田原・清研究室の学生および OB・OG に深く感謝の意を表する次第である。

2017年5月

執筆者一同

# 目 次

## 1章 歴史的背景

- 1.1 自律ソフトウェアとは ..... 2
- 1.2 エージェント技術による自律ソフトウェアの枠組みの登場 ..... 4
- 1.3 エージェント指向開発方法論 ..... 9
- 1.4 FIPA 標準 ..... 11
- 1.5 エージェント技術による自律ソフトウェアの進展 ..... 15

## 2章 自律ソフトウェア設計のためのエージェント技術

- 2.1 数理論理学：推論技術の基礎 ..... 23
- 2.2 プランニング：自己適応システムの基礎 ..... 26
- 2.3 ソフトウェア工学：自律ソフトウェア設計の基礎 ..... 30
- 2.4 自律 Web アプリケーションの要素技術 ..... 37

## 3章 ゴール指向要求工学

- 3.1 はじめに：要求工学の必要性 ..... 64
- 3.2 ゴール指向要求分析 ..... 65
- 3.3 ゴールモデル ..... 69
  - 3.3.1 ゴールとは ..... 69
  - 3.3.2 ゴールモデル ..... 71
- 3.4 ゴールのタイプとカテゴリ ..... 73
  - 3.4.1 ゴールタイプ ..... 73
  - 3.4.2 ゴールカテゴリ ..... 75



3.5	ゴールモデルの役割	78
3.6	形式的アプローチ	80
3.6.1	ゴール洗練化の意味論	80
3.6.2	洗練パターン	81
3.7	そのほかのゴール指向要求分析法	84
3.7.1	i*	85
3.7.2	NFR フレームワーク	88
3.8	ゴール指向要求工学と自律ソフトウェアとの関係	91
3.8.1	Gaia	91
3.8.2	Tropos	93
3.9	応用事例：ソフトウェア変更の局所化	95
3.9.1	ア プ ロ ー チ	96
3.9.2	ゴールモデルの整形	96
3.10	ま と め	100

## 4章 自己適応システム

4.1	はじめに：自己適応システムとは	102
4.2	自己適応システムの定義	102
4.3	自己適応システムの位置付け	104
4.4	自己適応システムの分類	106
4.5	主 要 モ デ ル	108
4.5.1	MAPE ループ	109
4.5.2	3 層アーキテクチャ	110
4.6	自己適応システムとエージェントとの関係	113
4.6.1	モデルの類似性	114
4.6.2	センサとエフェクタ	115
4.6.3	エージェント技術からのアプローチ	116
4.7	ソフトウェア工学分野からのアプローチ	117
4.7.1	ソフトウェアアーキテクチャ領域からのアプローチ	117

4.7.2	要求工学領域からのアプローチ	119
4.7.3	検証技術領域からのアプローチ	122
4.8	ケーススタディ：FUSION	124
4.8.1	モデル	126
4.8.2	適応サイクルと学習サイクル	128
4.8.3	実装環境	131
4.9	ケーススタディ：Zanshin	131
4.9.1	開発プロセス	133
4.9.2	実行例	134
4.10	まとめ	137

## 5章

## セマンティック Web エージェント

5.1	セマンティック Web とは	140
5.1.1	セマンティック Web の発展段階	141
5.1.2	メタデータ RDF	142
5.1.3	オントロジー OWL	156
5.1.4	検索言語 SPARQL	157
5.1.5	Linked Data	158
5.1.6	LOD	159
5.1.7	代表的な LOD	161
5.2	セマンティック Web エージェントの事例	164
5.2.1	質問応答システム	165
5.2.2	個人スケジュール管理	165
5.2.3	スマートシティ	166
5.2.4	イベント情報の統合	168
5.2.5	メディア情報の統合	169
5.2.6	モバイルセンサ活用	170
5.2.7	モバイル検索システム	172
5.2.8	セマンティック Web サービス	174

## 6章 自律 Web アプリケーション

6.1	自律 Web アプリケーションとは	177
6.2	情報抽出・予測	178
6.3	情報推薦	185
引用・参考文献		193
索	引	206

# 1 章

## 歴史的背景

### ◆本章のテーマ

マルチエージェントシステムは、近年のネットワーク上の大規模かつ複雑な分散システムやその上のサービス、およびシステムやサービスを利用する多数の人々の複雑な社会的振舞いに対し、設計や分析を効果的に可能とする手法として注目されている。しかしマルチエージェントシステム、あるいは一般にエージェントの概念と技術は、古くから多くの研究者が取り扱う研究分野として確立されており、現在隆盛を極めている本分野の諸技術につながっている。したがって、現在のマルチエージェントシステム技術の理解を深めるうえでも、歴史的背景を踏まえることはおおいに意義があるものと考えられる。本章では、そのような背景について解説する。

### ◆本章の構成（キーワード）

- 1.1 自律ソフトウェアとは  
エージェントの源流、エージェントの定義
- 1.2 エージェント技術による自律ソフトウェアの枠組みの登場  
BDI アーキテクチャ、言語行為論、KQML
- 1.3 エージェント指向開発方法論  
SWEBOK, Gaia, Tropos
- 1.4 FIPA 標準  
FIPA ACL, エージェント UML
- 1.5 エージェント技術による自律ソフトウェアの進展  
Telescript, Flage, Plangent, Bee-gent, JADE

### ◆本章を学ぶと以下の内容をマスターできます

- 📖 エージェントの定義
- 📖 エージェント指向開発
- 📖 FIPA 標準

## 1.1 自律ソフトウェアとは

本書において自律ソフトウェアとは、エージェントの自律性を特徴として活用しているソフトウェアであると定義する。自律性については、S. Franklin と A. Graesser によるつぎの定義<sup>67)</sup>†が、エージェントの分野で広く認められている。

「自律エージェントとは、環境の内部、またはその一部として存在するシステムであり、継続的に環境を監視し、環境に対し働きかけることにより、自分自身の目的を追求し、未来において監視できる環境の状態に影響を与えようとするものである」

したがって簡潔には、「自分の目的達成のために、環境とインタラクションを行うシステム（あるいはソフトウェア）である」といえる（スライド 1.1）。

### 自律ソフトウェアとは？

- 自律性：(ソフトウェア)エージェントの重要な特徴の1つ
  - [Franklin and Graesser '96]の定義
    - 環境とのインタラクション(環境への影響を含めて)
    - 自分自身の予定 (agenda, 意図, 目的)を追求
  - つまり、自分の目的達成のために、環境とインタラクションを行うシステム(あるいはソフトウェア)

スライド 1.1 自律ソフトウェアとは

† 肩付数字は巻末の引用・参考文献を示す。

なお、エージェントの概念は情報技術関連の幅広い研究分野において多くの研究者により利用されている。そのため、エージェントの定義や説明は研究者により多様なものとなっている。例えば S. Franklin と A. Graesser は、前述の定義を提起するにあたり、10 件もの定義を参考に行っている<sup>67)</sup>。ここでは、そのほかのエージェント概念の説明としてつぎの3点を取り上げる(スライド 1.2)。「パーソナルコンピュータの父」と呼ばれるアラン・ケイ (A. Kay)<sup>80)</sup> は、「エージェントは計算機の世界の中に住んで仕事を行う『ソフトロボット』であろう」と述べている。K. Crowston と T.W. Malone<sup>51)</sup> は、「(人間の協調プロセスにおける) 各役割に対し、メッセージの集合の全体、または一部を自動的に処理することにより支援するエージェントを、われわれは想像できる」と記している。さらに P. Maes<sup>91)</sup> は、「エージェントに対し『ユーザと同じ作業環境で協調するパーソナルアシスタント』というメタファを使用する」としている。

## エージェントとは？

- 研究者により多様な定義・説明
- アラン・ケイ ([Kay '84]) : 計算機の世界の中に住んで仕事を行う「ソフトロボット」
- [Crowston and Malone '88] : メッセージの集合の全体、または一部を自動的に処理することにより、人間の協調プロセスを支援
- [Maes '94] : ユーザと同じ作業環境で協調するパーソナルアシスタント

## 1.2 エージェント技術による自律ソフトウェアの枠組みの登場

エージェント技術を用いたさまざまな自律ソフトウェアが開発、提案されるにつれて、多くの問題点の認識や知見が蓄積された（スライド 1.3）。最大の問題点は、自律ソフトウェアの構造や振舞いが複雑となることである。その理由の1つとして、エージェントはオブジェクト指向と人工知能技術を活用しているため、その両者の複雑さが組み合わされているという点が挙げられる。またエージェントと環境とのインタラクションにおいて、環境が予期せぬ変化を起こすことを考慮しなければならないことが第2の理由である。このような問題を解決するための知見として、エージェントアーキテクチャ、エージェント間インタラクション、およびエージェントの各種の特性の研究が進められた。その結果、これらの研究成果はエージェント技術による自律ソフトウェアの枠組みとして体系化されるに至っている。

### 背景

- 多数の自律ソフトウェアの開発・提案  
→問題点の認識や知見の蓄積が進行
  - 問題点：構造や振舞いが複雑
    - オブジェクト指向+人工知能技術
    - 環境とのインタラクション→環境は予期せぬ変化を起こしうる
  - 知見：多くの共通概念・理論・技術
    - エージェントアーキテクチャ：BDIなど
    - エージェント間インタラクション：言語行為論, KQMLなど
    - エージェントの特性：リアクティブ・プロアクティブ, 知性, 移動性, 学習, 協調, 適応, 創発など
- エージェント技術による自律ソフトウェアの枠組みとして体系化

このような枠組みの1つとして、まず**BDI**アーキテクチャを取り上げる（スライド 1.4）。BDIアーキテクチャは、M. Bratman<sup>43)</sup>が提唱したエージェントの意図に関する理論に基づき、P.R. Cohen, H.J. Levesque<sup>50)</sup>、およびM.P. Georgeffら<sup>110)</sup>が、単一のエージェントのアーキテクチャへと発展させたものである。本アーキテクチャは、エージェントの状態を人間の心的状態になぞらえ、つぎの3つのものを主要な構成要素とする。

- 信念 (Belief) : (各時点で) エージェントが信じている (環境や自分自身の) 状態
- 欲求 (Desire) : エージェントが達成したいと思っている状態
- 意図 (Intention) : 欲求を達成するために実際に実行する行動

そのうえで、エージェントは環境や自分自身を監視して、得られた情報に基づき信念を更新し、欲求を満たすための意図を推論して実行する。

実際のシステムでは、BDIアーキテクチャの構成はスライド 1.5 のようにな

## BDI アーキテクチャ

- 個々のエージェントのアーキテクチャ
- M. Bratman, P.R. Cohen, H.J. Levesque, M.P. Georgeff らが提唱
- BDI: Belief, Desire, Intention (信念, 欲求, 意図)
  - Belief : (各時点で) エージェントが信じている (環境や自分自身の) 状態
  - Desire : エージェントが達成したいと思っている状態
  - Intention : Desireを達成するために実際に実行する行動



# 索

# 引

<b>【あ】</b>	
アクタ	65
アプリケーション ロジック	106
蟻コロニー最適化	52
安全性	76
<b>【い】</b>	
一階述語論理	23
一貫性	81
遺伝アルゴリズム	50
意味論	23
インタフェース	76
インタラクション	91
<b>【え】</b>	
エージェント	2, 69
エージェント間インタ ラクション	7
エージェント指向開発 方法論	10
エージェント指向ソフト ウェア工学	91
エージェントモデル	68
エフェクタ	115
<b>【お】</b>	
オートノミックコン ピューティング	104
オブジェクト	32
オブジェクト指向開発 方法論	32
オブジェクトモデル	68
オープンデータ	159
オントロジー	140
オンライン学習	130

<b>【か】</b>	
階層プランニング	29
外部アプローチ	107, 132
係り受け解析	56
拡張 Darwin モデル	117
確率論	37
可用性	76
環境アクタ	72
監視	109
完全性	76, 81
<b>【き】</b>	
機械学習	37
記号論理学	23
記述の表現	70
記述論理	23
期待	68, 73
機能ゴール	75
規範の表現	69
機密性	76
強化学習	47
協調フィルタリング	53
強適応	106
<b>【く】</b>	
クラス	32
クラスタ性	60
クラスタリング	42
<b>【け】</b>	
計 画	109
形式検証	35
形式手法	35
形式仕様記述	35
継続的なソフトウェア 進化	100
形態素解析	55
決 定	96

決定木	39
言語行為論	7
原子命題	23
<b>【こ】</b>	
効 果	27
貢献リンク	70, 87
勾配法	49
効 用	128
コネクタ	119
ゴール	26, 69
ゴールカテゴリ	75
ゴール管理層	110
ゴール指向要求分析	65
ゴールタイプ	73
ゴールモデル	67
コントローラ	109
コンフィギュレーション	111
コンポーネント	110, 117, 119
コンポーネント管理層	111
コンポーネントモデル	117
<b>【さ】</b>	
最小性	81
最適化手法	48
サブサンクション アーキテクチャ	115
サポートベクターマシン	40
<b>【し】</b>	
自己管理	104
自己構成	104
自己最適化	104
自己修復	104
自己適応システム	96, 102
自己適応性	102
自己適応ソフトウェア	102
自己認識	105

自己防衛	104	洗練パターン	81	ナイーブベイズ	43
自然言語処理	55			内容ベースフィルタ	
事前条件	27	<b>【そ】</b>		リング	54
時相論理	23	操作モデル	68	<b>【に】</b>	
実行	96, 109	測定基準	128	日本語 Wikipedia	
弱適応	106	組織	92	オントロジー	162
収集	96	ソフトウェアアーキ		ニューラルネットワーク	40
充足可能性	78	テクチャ	33	認識要求	120
受益者	85	ソフトウェア開発方法論	10	<b>【は】</b>	
手段目的リンク	86	ソフトウェア工学	30		
条件付き確率場	46	ソフトウェア進化	80, 96	パフォーマンスティブ	7
状態	26	ソフトゴール	75	<b>【ひ】</b>	
状態爆発	122	<b>【た】</b>		非機能ゴール	75, 88
初期状態	26	タスク分解リンク	86	非線形プランニング	27
自律エージェント	2	<b>【ち】</b>		開いた適応	108
自律性	2	知識	110	<b>【ふ】</b>	
自律ソフトウェア	2	<b>【つ】</b>		ファシリテータ	
自律 Web アプリ		追跡可能性	80	エージェント	8
ケースション	177	<b>【て】</b>		フィーチャ	124, 126
信頼性	76	提供者	85	フィードバック制御	
<b>【す】</b>		ディープニューラル		ループ	114
推論規則	23	ネットワーク	42	複合命題	24
数理計画法	48	ディープラーニング	42	不確かさ	121
数理論理学	23	定理証明	24	ブラックボックス	
スケールフリー性	60	適応	96	アプローチ	106
スコープ	78	適応エンジン	107	プラン	26, 112
ステークホルダ	64	適応ロジック	106	プランクノード	146
スモールワールド性	60	適応ロジック	106	プランニング	26, 113
<b>【せ】</b>		テストイング	122	振舞いゴール	73
制御変数	120	<b>【と】</b>		フレームワーク	124
制御ループ	96	統計学	37	プロアクティブな適応	108
性能	76	動的検証	122	プロパティ	144
責務モデル	68	閉じた適応	108	分解リンク	70
セキュリティ	76	トピック分析	58	分析	96, 109
セマンティック Web	140	トリプル	144	文脈認識	105
セマンティック Web		トレーサビリティ	80	<b>【へ】</b>	
サービス	174	<b>【な】</b>		バイジャンネットワーク	43
線形計画法	49	内部アプローチ	106	ページランク	61
線形プランニング	27			変更管理層	111
センサ	115				
洗練化リンク	70				

<b>【ほ】</b>		<b>【も】</b>		<b>【り】</b>	
ホワイトボックス		目的ベースエージェント	109	リアクティブな適応	108
アプローチ	106	モデル検査	24, 122	リアクティブ プランニング	27
<b>【ま】</b>		<b>【や】</b>		利害関係者	64
マルコフネットワーク	45	焼きなまし法	51	離散時間マルコフ連鎖 モデル	123
マルコフ論理 ネットワーク	45	<b>【よ】</b>		リソース	144
満足化ゴール	88	要 求	64, 73	<b>【ろ】</b>	
満足度	128	——の監視	119	ロール	91
<b>【め】</b>		——の完全性	79	論証ゴール	88
命 題	23	要求獲得	64	論理式	23
命題論理	23	要求管理	64	<b>【わ】</b>	
命題論理演算子	24	要求記述	64	ワトソン	165
メタデータ	140	要求検証	64		
メトリクス	128	要求工学	64		

<b>【A】</b>		availability	76	confidentiality	76
ABLE	116	Avoid ゴール	74	configuration	111
Achieve ゴール	73	AwarenessRequirement	120	consistency	81
ACO	52	AwReq	120	context-awareness	105
Act	96	<b>【B】</b>		continuous software evolution	100
adaptation	96	BDI アーキテクチャ	5	contribution links	70
adaptation engine	107	BDI エージェント	94	Control loop	96
Adapter パターン	116	Bee-gent	18	Control loop パターン	98
agent	69	behavioral goals	73	control variable	120
Agent Building and Learning Environment	116	black-box approach	106	CRF	46
agent model	68	<b>【C】</b>		C2	119
Agent UML	12	Cease ゴール	74	<b>【D】</b>	
Agent-Oriented Software Engineering	91	change management layer	111	DBpedia	161
Analyze	96, 109	Chaos レポート	64	Decide	96
AND-洗練化リンク	71	C.I.A.	76	decomposition link	70
AOSE	91	Collect	96	dependee	85
argumentation goals	88	completeness	81	depender	85
Autonomic Computing	104	component control layer	111	discrete time Markov chain model	123
				Dublin Core	149

<b>[E]</b>		Java Management		<b>[O]</b>	
Eclipse Modeling Framework	133	eXtension	116	object model	68
EMF	133	JMX	116	operation model	68
environmental actors	72	JSON-LD	152	organization	92
Execute	109	J-GLOBAL knowledge	162	OR-洗練化リンク	72
expectation	68	<b>[K]</b>		OWL	156
external approach	107	KAOS	65	<b>[P]</b>	
<b>[F]</b>		Knowledge	110	Plan	109
feature	126	KQML	7	Plangent	17
feedback control loop	114	<b>[L]</b>		P/N 判定	58
FIPA	12	LDA	58	Prometheus	94
FIPA ACL	12, 14	Lined Open Data	159	Proxy パターン	116
Flage	16	Linked Data	159	<b>[R]</b>	
functional goals	75	Linked Open Social Signals	179	Rainbow	108
FUSION	106, 124	<b>[M]</b>		RDF	142
<b>[G]</b>		Maintain ゴール	74	RDF スキーマ	148
GA	50	MAPE ループ	104, 109	RDFa	154
Gaia	91	MAPE loop	109	RDF/XML	155
goal	69	MAPE-K ループ	110	refinement links	70
goal management layer	110	MaSE	94	refinement pattern	81
goal model	67	Meta-Object-Facility	133	RELAX	121
goal-oriented requirements analysis	65	metrics	128	requirements	64
GoF のデザインパターン	116	Milestone-driven 洗練パターン	82	requirements engineering	64
Guard-introduction 洗練パターン	82	minimality	81	responsibility model	68
<b>[H]</b>		MLN	45	role	91
HITS	61	model checking	122	<b>[S]</b>	
<b>[I]</b>		MOF	133	SA	51
i*	85, 93	Monitor	109	satisficing goals	88
integrity	76	<b>[N]</b>		schema.org	149
interaction	91	NFR フレームワーク	88	SD モデル	85
internal approach	106	NFR goals	88	SecureTropos	94
IRI	145	NonFunctional Requirements goals	88	self-adaptiveness	102
<b>[J]</b>		non-functional goals	75	self-adaptive software	102
JADE	19, 116	N-Triples	151	self-adaptive systems	102
				self-awareness	105
				self-configuration	104
				self-healing	104
				self-management	104
				self-optimization	104

self-protection	104	SWEBOK	30		
self* システム	105	system to be	72		
self* systems	105				<b>【W】</b>
software evolution	80				weak adaptation 106
soft goals	75	Telescript	15		Web 177
SPARQL	157	testing	122		white-box approach 106
SR モデル	85	TF-IDF	57		World Wide Web 177
StarMX	116	traceability	80		WWW 177
STAR-CITY	166	Tropos	93		W3C 141
Strategic Dependency モデル	85	Turtle	151		
Strategic Rationale モデル	85				<b>【Z】</b>
Strategy パターン	116	UML	32, 33		Zanshin 108, 131
strong adaptation	106	utility	128		Zanshin フレームワーク 116, 121
subsumption architecture	115				<b>【数字】</b>
SVM	40	variation point	120		3 層アーキテクチャ 110
		VP	120		3 層アーキテクチャ モデル 131

—— 著者略歴 ——

大須賀昭彦（おおすが あきひこ）

1981年 上智大学理工学部数学科卒業  
1981年 東京芝浦電気株式会社勤務  
1985年 新世代コンピュータ技術開発機構  
～89年 出向  
1995年 工学博士（早稲田大学）  
2007年 電気通信大学教授  
現在に至る

田原 康之（たはら やすゆき）

1989年 東京大学理学部数学科卒業  
1991年 東京大学大学院理学系研究科修士課程  
修了（数学専攻）  
1991年 株式会社東芝勤務  
2003年 国立情報学研究所勤務  
2004年 博士（情報科学）（早稲田大学）  
2008年 電気通信大学准教授  
現在に至る

中川 博之（なかがわ ひろゆき）

1997年 大阪大学基礎工学部情報工学科卒業  
1997年 鹿島建設株式会社勤務  
2007年 東京大学大学院情報理工学研究科修士  
課程修了（創造情報学専攻）  
2008年 東京大学大学院情報理工学研究科博士  
課程中退（創造情報学専攻）  
2008年 電気通信大学大学院情報システム学研  
究科助教  
2013年 博士（工学）（早稲田大学）  
2014年 大阪大学大学院情報科学研究科准教授  
現在に至る

川村 隆浩（かわむら たかひろ）

1992年 早稲田大学理工学部電気工学科卒業  
1994年 早稲田大学大学院理工学研究科修士課  
程修了（電気工学専攻）  
1994年 株式会社東芝勤務  
2001年 博士（工学）（早稲田大学）  
2001年 米国カーネギーメロン大学客員研究員  
～02年  
2003年 電気通信大学大学院客員准教授  
2007年 大阪大学大学院非常勤講師  
2015年 国立研究開発法人科学技術振興機構情  
報分析室主任調査員  
現在に至る

# マルチエージェントによる自律ソフトウェア設計・開発

Intelligent Software Design and Development Based on Multi-Agent Technology

© Osuga, Tahara, Nakagawa, Kawamura 2017

2017年7月21日 初版第1刷発行

検印省略

著者 大須賀 昭彦  
田原 康之  
中川 博之  
川村 隆浩  
発行者 株式会社 コロナ社  
代表者 牛来 真也  
印刷所 三美印刷株式会社  
製本所 有限会社 愛千製本所

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 978-4-339-02818-8 C3355 Printed in Japan

(新井)



**JCCOPY** <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。