

Flageアーキテクチャの構想†

4 N-4

桑野 文洋¹来間 啓伸²蓬萊 尚幸³松浦 佐江子⁴本位田真一⁵

情報処理振興事業協会(IPA)
新ソフトウェア構造化モデル研究本部

1 はじめに

環境の変化に柔軟に適應するソフトウェアを構築するためのソフトウェア・アーキテクチャ、Flageアーキテクチャについて述べる。変化に“柔軟なソフトウェア”という言葉からは、たとえば、ソフトウェアの変更や修正のやりやすさといった特徴を連想するが、本研究においてはもう一步進んだ目的を有している。それは、ソフトウェア・システムが置かれている環境が変化した場合においても、ソフトウェア・システム自身が極力人間の手を借りずに、その変化に追従できるようにしようというものである。ここで意味するソフトウェアにとっての環境とはソフトウェアに対する要求のことを指す。要求とはサービスを提供する機能としての要求と稼働条件（動作環境）としての要求のことである。

開放型ネットワークにおいては、ソフトウェア・システムは不特定のユーザやプロセスからさまざまな要求を受けることが想定され、しかもその変化のタイミングおよび内容は事前に特定できない。こうした変化に対し、稼働時に動的かつ自動的に適應したり、そうでなくとも極力人手を労さずに適應できる柔軟なソフトウェア（適応型ソフトウェア）を構築することは、開放型環境において、柔軟かつ安定的な情報サービスをユーザに提供することにつながる。

一方、開放型ネットワークにおいては、フリーソフトウェアを含め多くの情報がネットワーク上で公開されており、ユーザはそれらを自由に転送し、利用することができる。ソフトウェア開発においては、既存のソフトウェア部品の再利用による生産性の向上が試みられているが、こうしたネットワーク上でのソフトウェアの公開・利用形態はソフトウェア再利用による開発形態の1つの可能性を示唆している。

本研究では、上記の点に着目し、自律的にネットワーク上を渡り歩き、そこに存在するソフトウェアを獲得することで、要求の変化に対応するオブジェクト（エージェント）を導入する。Flageアーキテクチャでは、複数のエージェントと場と呼ばれるオブジェクトから適応型ソフトウェア・

システムを構成する。本稿では、こうした自律的かつ柔軟なエージェント・モデルを明らかにした上で本アーキテクチャを概説する。

2 エージェント・モデル

エージェントとは、要求の変化に適應するために、自分の自身の機能を動的に変更できるモジュールであり、たとえば、オブジェクト指向計算で言うオブジェクトの自律性を強めたような活動主体であると考えてよい。システムは複数のエージェントから構成され、各エージェントは、各自が構成要素として存在する環境において、それぞれ並行に動作する。エージェントを独立並行に動作する実行主体とした理由は、提供する各サービスを効率的に実行し、部分的に機能が停止してもシステム全体への影響を最小限にする分散システムを開発の対象としているためである。

環境には、エージェントが守らなければならない規則やエージェントが共通に利用する機能（プログラム）およびその他の情報が存在する。エージェント間の規則や共有情報は、複数のエージェントが互いに競合なく動作し、また要求の変化などに対して、各自が持っている情報を互いに交換しながら協調し、対応するためのものである。こうした環境を表現するために場を導入する。場はネットワーク内で各エージェントが訪れる場所でもあり、場に入ることによってエージェントはそこに存在する機能や情報を得ることができる。場はネットワーク内の複数のノード上で開発者やエージェントによって生成される。

エージェントは、要求の変化を認識すると、たとえば未知のメッセージを受信すると、エージェント自身（またはそのクローン）がその要求を満たす機能が存在する場を求めて、探し回る。そして各場で機能を獲得し、変化した要求に対応できるよう成長することで適應を行なう。このような変化した要求に対するエージェントの適應のプロセスを自己形成プロセスと呼ぶ。エージェントは適應に成功した場合、他のエージェントがその自己形成プロセスを再利用できるよう、エージェント間の共有情報としてそれを場に記録することができる。

エージェントは、結果的に、分散環境上で開発されたすべての機能を場を媒体として利用できる。したがって、変化の適應もその範囲内で可能となる。ただし、分散環境が構築されているネットワークの開放性により、この範囲は常に拡大する可能性がある。エージェントに場を自律的に渡り歩く機能を与えることにより、こうした分散環境下で、

†Flage Architecture - Conception

Fumihiro Kumeno, Hironobu Kuruma, Hisayuki Horai, Saeko Matsuura, Shinichi Honiden

¹(株)三菱総合研究所より出向中²日立より出向中³富士通研究所より出向中⁴管理工学研究所より出向中⁵現 東芝

エージェントは変化への適応を各々独立並行に行なうことができる。ここで意味するエージェントの自律性とは、エージェントが以下の機能を有することを指す。

- エージェント自身で、要求を満たす機能を求め、行き先の場を探し、移動する。
- 行き先を探すための場の情報を得るため、不特定多数のエージェントと情報交換を行なう。
- 場にある機能が要求を満たすものかどうかを見極める手段を持っている。
- 獲得した機能とエージェントとすでに持っている機能との間のインタフェースに不整合があるときは、これを修正して自らの機能として取り込む。

変化に柔軟なエージェントの要件をまとめると、以下のようになる。

1. 自分自身の機能をデータとして扱うことができ、動的に変更できる。
2. 場を自律的に渡り歩くことができ、必要な機能や情報を獲得することができる。
3. 自己形成プロセスを自身内に記録し、再利用するメカニズムを有することができる。さらにそれらの情報を場に公開することができる。

従来のオブジェクト指向では、オブジェクトが上記の要件を満たさないため、適応型ソフトウェアの構築が難しい。こうした問題点を解決を目指し、適応型ソフトウェア構築のプラットフォームの提供を目標としたソフトウェア・アーキテクチャとしてFlageアーキテクチャを提案している。

3 Flageアーキテクチャ

Flageアーキテクチャはカーネル言語と適応システムを適応型ソフトウェア構築のプラットフォームとして持つ(図1)。

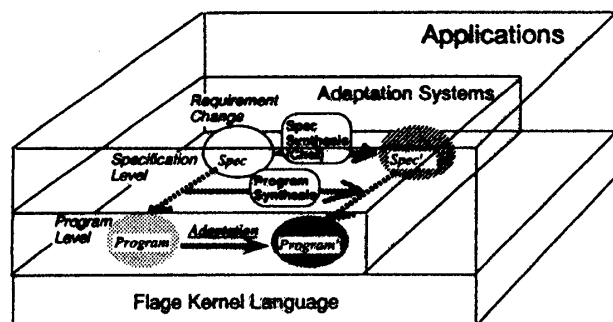


図 1: Flageアーキテクチャ

Flageカーネル言語 エージェントや場を定義するためのプログラミング言語であり、アーキテクチャの基本機能を提供する。言語は並行オブジェクト指向モデルをベースとし、

前章の1や2の要件を満たすべく、メタ階層を有し、ネットワーク間を移動できるオブジェクト(エージェント)とエージェントが出入りを行なう場を記述する機能を提供している。

適応システム 適応システムは前章の3にあげたエージェントの機能を実現するために提供される機能である。特に単に場を渡り歩いてプログラムを獲得するといった適応手段よりも高度な適応手段をエージェントに与えるエンジンとなる。適応システムには仕様レベルでの適応機能とプログラムレベルまでの適応機能が存在する。すなわち、既存の仕様と仕様に対する変更要求から、その変更要求を満たす新たな仕様を合成する仕様生成機能(Spec Synthesis)と、既存のプログラム(仕様およびその開発プロセスを含む)と変更した仕様からプログラムを合成する機能(Program Synthesis)が存在する。両機能ともにFlageカーネル言語上のアプリケーションシステムとして提供されるが、そこで利用される仕様記述言語およびプログラム記述言語はFlageカーネル言語には限定されない。

4 おわりに

本稿では環境の変化に柔軟な適応型ソフトウェアを構築するソフトウェア・アーキテクチャ、Flageアーキテクチャのコンセプトを説明した。本アーキテクチャを構成する各機能(Flageカーネル言語、適応システム)は、現在独立に並行して開発が進んでいるが、将来的には統合される予定である。本アーキテクチャ上で開発されるアプリケーションの問題領域としては、コンポーネントウェア[3]に代表されるように、再利用部品による合成によって開発されるアプリケーションを対象としている。場に再利用部品を置き、自律的に場を渡り歩き、必要に応じて部品を取捨選択するエージェントを用意することにより、複合オブジェクトより変化に柔軟な機能のソフトウェアを実現できる。

謝辞 本研究は、産業科学技術研究開発制度「新ソフトウェア構造化モデルの研究開発」の一環として情報処理振興事業協会 (IPA) が新エネルギー・産業技術総合開発機構から委託をうけて実施したものである。

参考文献

- [1] 本位田 真一: “協調型ソフトウェアアーキテクチャ”, 日本ソフトウェア科学会チュートリアル資料, 分散人工知能-協調計算とマルチエージェント・システム-, 1992.
- [2] 桑野 文洋 他: “Field Oriented Language, Flage”, ソフトウェア工学の基礎ワークショップ(FOSE'94), 日本ソフトウェア科学会, 1994.
- [3] 青山 幹雄: “コンポーネントウェアと協調オブジェクトの動向”, オブジェクト指向'95 シンポジウム, 情報処理学会, 1995.