動的適応可能なソフトウェアの系統的構成方式と言語の研究開発

天野 憲樹
北陸先端科学技術大学院大学 情報科学研究科

平成 11 年 2 月

論文の内容の要旨

開放型分散システム環境や移動計算機環境の普及にともない，プログラマは設計の段階でソフトウェアの実行環境およびその動的な状態変化を完全に予測することができなくなった．しかし，多様な実行環境とその動的な状態変化がプログラムの実行に与える影響は大きく，これを無視することはできない．それゆえ，今日のソフトウェアは多様な実行環境で動作し，その動的な状態変化に適応できることが要求される．本研究では，このようなソフトウェアを指して動的適応可能なソフトウェアと呼ぶ．一般に，こうした動的適応可能なソフトウェアを作成することは困難であり，その有効な構成手法が望まれている．

本研究では，動的適応可能なソフトウェアのためのモデル **DAS** を提案し，DAS モデルにもとづくプログラミング言語の基本機構を設計・開発する．DAS モデルは，さまざまな実行環境やその動的な状態変化に適応する機構を持ち，DAS モデルにもとづくプログラミング言語は，動的適応可能なソフトウェアの柔軟な記述を支援する．本研究では，こうした DAS モデルおよびその記述言語の有効性を示すとともに，DAS モデルにもとづく動的適応可能なソフトウェアの有効性を示す．

動的適応可能なソフトウェアを実現する上で，各ソフトウェア自身の振舞いを実行環境の状態に応じて変えることは有効な方法である．しかし，言うまでもなく同一のソフトウェアに対して実行環境ないしその状態ごとに振舞いの異なるバージョンを多数作成することは現実的ではない．その上，実行環境やその状態変化に対するソフトウェアの振舞いは，各ソフトウェアの性質，さらにはその個々の処理の性質に依存する場合が多く，これを外部から汎用的に制御することは困難である．このため，各ソフトウェア自身が実行環境やその動的な状態変化に対処（適応）する能力を持つ必要がある．本研究では，このソフトウェアの能力を指して動的適応可能性と呼ぶ．適応適応可能なソフトウェア，すなわち動的適応可能性を持つソフトウェアは，さまざまな実行環境やその動的な状態変化に適応できるだけでなく，その適応動作により各実行環境固有の特性を最大限に活かすことができるようになると考えられる．

しかし，プログラマが事前に予測可能な実行環境とその状態には限界があるため，プログラマが事前にソフトウェアに与えることができる動的適応可能性にもやはり限界がある．このため，ソフトウェアの動的適応可能性は，拡張が可能な構造で実現され，事前に予測できなかった実行環境やその動的な状態変化に対して，拡張/変更できる必要がある．

本研究では，そのような拡張可能な動的適応可能性を備えたソフトウェアのモデル DAS (以下，DAS モデル) を提案する．以下は，DAS モデルの主要な概念である．

- ソフトウェアにおける適応動作の単位を手続きとし，手続き呼び出しにもとづいて適応動作の制御を行なう．
- DAS モデルは，適応可能手続きと呼ばれる実行環境の状態にもとづく総称手続き (総称関数) を持つ．
- 適応可能手続きの制御機構は，自己反映計算を利用して実現される．
- DAS モデルにもとづく動的適応可能なソフトウェアは，メタレベルアーキテクチャを形成する．

これらの概念は，本研究の以下の目標にもとづいている．

- 既存言語によるソフトウェアへの高い親和性．
- 柔軟性の高い動的適応可能性の実現．
- 高い可読性，保守性，拡張性の実現．

既存のソフトウェア資産を開放型分散システム環境や移動計算機環境で有効に利用したいという要求は大きい．それゆえ，既存のプログラミング言語で記述されたさまざまなソフトウェアに動的適応可能性を導入できる方式が望ましい．DAS モデルでは，ソフトウェアにおける適応動作の単位を手続きとし，ソフトウェアの適応動作は適応可能手続きと呼ばれる実行環境の状態にもとづく総称手続き (総称関数) として実現される．ひとつの適応可能手続きは，実行環境の状態に応じた実行コードを持つ複数の適応可能メソッドから構成される．適応可能手続きが呼び出されると，実行環境の状態に応じて適切な適応可能メソッドが選択され，実行される．この手法は，原理的に手続き呼び出しの機構を持つ多くの既存言語により記述されたソフトウェアに適用できる．この結果，既存のソフトウェア資産を有効利用できる可能性が高まる．本研究では，DAS モデルの基本機構を Java 言語のクラスライブラリ (以下，DAS クラスライブラリ) として設計・実装した．そして，実際に DAS クラスライブラリを用いて Java のアプリケーションに動的適応可能性の導入を行なった．これらの実装を通じて，DAS モデルが既存言語およびそのソフトウェア対して高い親和性を持つことを実証した．

本研究の手法において，ソフトウェアの柔軟な動的適応可能性を実現するには，適応可能手続きの制御機構を実行環境の状態に応じて柔軟に変更できることが求められる．DAS モデルでは，この制御機構自体が適応可能手続きを用いて自己反映的に実現される．そのため，この制御機構自体も実行環境の状態変化に適応することができる (実行環境の状態変化に応じて変化する)．実行環境の状態変化に応じて，この制御機構を変えることで，ソフトウェアの柔軟な適応動作を実現することが可能となる．たとえば，実行環境の状態に応じたソフトウェアの機能の一時的な制限やその解除といった適応動作を，プログラムを修正することなく実現することができるようになる． さらにこの制御機構が，適応可能手続きを用いて実現されるため，この機構のカスタマイズは，その構造を修正するこ

となく，適応可能メソッドのインクリメンタルな追加で実現できる．これらの利点について，DAS クラスライブラリを用いた例題の実装を通して，実際に確かめることができた．また，ソフトウェアにおける柔軟な適応動作の有効性も実際に例題を用いて示した．

DAS モデルにもとづく動的適応可能なソフトウェアは，メタレベルアーキテクチャを形成する．そのメタレベルは，ソフトウェアにおける適応動作の制御を実現する機構の記述からなり，ベースレベルは，各ソフトウェアにおける問題領域の記述からなる．つまり，適応動作の制御機構は，ベースレベルの記述から完全に分離され，メタレベルに隠蔽される．この記述の分離は，動的適応可能なソフトウェアの可読性，保守性，拡張性を向上させる．この利点についても例題の実装を通して，実際に確かめることができた．

キーワード: 動的適応，自己反映計算，適応可能手続き，開放型分散システム環境，移動計算機環境