

3 当初の研究計画

アクティブソフトウェア (Active Software) の4つの特徴の中で注目した

- (a) 自分の状況を知る (self awareness) と、
 - (d) 能動的に移動する (mobile) を実現するために、
- 3つの研究目的に沿って、研究を進める。

3.1 平成15年度の研究計画・方法

(1.1) アクティブソフトウェアの設計と記述：

アクティブソフトウェアの構成は、計算状況を監視して、期待される計算状態に対応した処理を指示することが基本になる。

それを記述するために、これまでは、「起動条件」と「関数本体」をもつ「能動関数」を導入してきた。

具体的な問題を取上げて、アクティブソフトウェアの構成を検討してきた結果、

- (n1) 機能が類似した能動関数の配列を導入すること、
- (n2) 能動関数が分散環境で稼働すること、
- (n3) 予想の難しい異常状態に能動関数を適用すること、

等が有用で有ることが判明した。これに対応する記述法を導入して、新しいアルゴリズムの設計法を考察する。

(1.2) アクティブソフトウェアの開発環境の構成：

設計されたアクティブソフトウェアを、実行形式に変換する言語プロセッサの構成を考える。

ある状態が検出されたとき、起動する能動関数は1つとは限らない。実行環境とソフトウェアの要求に応じて、逐次実行、並列実行、分散実行に対応できる言語プロセッサの構成法が問題となる。

また、新しい記述法の導入 (たとえば上の n1, n2, n3) に対応して、言語プロセッサをどのように柔軟に拡張していくかを考慮する。

(1.3) アクティブソフトウェアを実行する分散計算環境：

ネットワーク内での分散計算環境で、アクティブソフトウェアの全体、あるいは、一部の能動関数の移動性 (mobility) を実現する準備をする。

能動的な移動の判断を含めて、移動のタイミング、移動におけるプロトコル、移動先での実行の再開などの方法を検討する。

3.2 平成16年度と平成17年度の研究計画・方法

平成15年度までに得た準備の結果を基に、具体的な大きな問題をアクティブソフトウェアとして設計し、システムの改善を計り、将来の発展への方向を探る。

(2.1) アクティブソフトウェアの設計と記述：

アクティブソフトウェアに適した問題のアルゴリズムを考えて、記述形式の改善と収束を計る。アクティブソフトウェアには、並列性の有る問題、局所的な関係で処理を決めやすい問題 (たとえば、並列構文解析法、グラフの描画・グラフ探索、量子計算アルゴリズムなど) や、異常状態をもつ問題などが適している。

状況によっては行動を変更する場合もある。このような柔軟な前提での、アルゴリズムの考え方と記述形式を考察する。併せて、設計の考えやすさ・柔軟性、および、実行速度などの点から、アルゴリズムの評価法を求める。

(2.2) アクティブソフトウェアの開発環境の構成：

言語プロセッサには、記述法の変更や拡張に適切に対応すること、移動性や動的結合など分散環境での実行に適した出力形態を生成すること、の機能が要求される。言語プロセッサを柔軟に作成する言語プロセッサ生成系の構成を考える。

(2.3) アクティブソフトウェアを実行する分散計算環境：

ソフトウェアの動的な移動と変更を実現し、分散計算の並列性の効果を評価する。一方、通信による応答遅延の影響を調べて、応答時間に依存しない実現法を考える。計算量が増大する量子計算アルゴリズムの実装の可能性を探る。

3年間の研究を通じて、アクティブソフトウェアへのパラダイムシフトへの土台を築く。これは、人の思考の形態を忠実に実現し、柔軟なプログラムの構成のひとつの方式を与えるとともに、新しいソフトウェアとハードウェアの構成に良い効果を及ぼすと考えている。

本研究は、ソフトウェアの設計と実装において、「ソフトウェアは能動的に動き出す処理の集まりである」という考えに基づいている点に特色がある。状態による能動的な人の行動や思考の形態に合わせているので、ソフトウェアを柔軟で拡張しやすくするとともに、信頼性を向上させるものである。

3.3 実施に当たっての各年度の研究目標と実施計画

当初は、特徴 (a. 自分の状況を知る)、特徴 (d. 動き回る) に重点を置いた。しかし、次の2点から、研究の方針を若干を変更して、各年度の研究目標と実施計画に具体性を加味して変更した。

1. 研究を実施する平成15年度に入って、研究組織が強化されて、研究計画の立案時点から変わった。
2. ソフトウェアを取り巻く問題の変化から、特徴 (b. 自分を変更する)、および、それに関連して「安全に走行する」ことが重要である。

3.3.1 平成15年度の研究の目標・研究計画

平成15年度の研究の目標

新しい情報システムの設計に当たっては、機能の高度化・大規模化、開発期間の短縮化が要求されている。それに伴って、設計・開発の時点で仕様が確定しなかったり、使用環境の変化に伴い要求仕様が変更されたりすることがある。また、処理が分散してソフトウェアの移動性が求められている。

本研究では、これらの問題に対して、アクティブソフトウェア (Active Software) の考えに基づいて、安全に働く新しい大規模なシステムの設計法を確立することを目指している。その基盤として、アクティブソフトウェアを開発して実行できる環境を構築することを目的としている。

具体的には次の3つの面から実現に関する研究を進める。

- (1) ソフトウェアの設計過程を分析して、アクティブソフトウェアに適した設計方法を確立する。
- (2) アクティブソフトウェアの基本的な開発環境として、アクティブソフトウェアを記述する方法とそれを実現する言語プロセッサを構成する。

(3) アクティブソフトウェアを分散計算環境で実行する形態や、アクティブソフトウェアを実行するのに適した新しいコンピュータアーキテクチャの構成など、効率の良い実行形式を統合的に考察する。

「ソフトウェアは能動的に動き出す処理の集まりである」という見方のソフトウェア設計方法の確立を目指す。

平成 15 年度の研究実施計画

アクティブソフトウェアに関連してこれまでに能動関数と能動形プログラミング、および、自発性をもつオブジェクトを含めたオブジェクト指向設計などの研究を進めてきた。

平成 15 年度はこれらを基にして、上に示した研究の目標に向けて、次の点で研究を始める。

(1) アクティブソフトウェアに適したアルゴリズムの開発と設計法 :

アクティブソフトウェアの構成は、計算状況を監視して、期待される計算状態に対応した処理を指示すること(何時、どんなときに、何をするか)が基本になる。このような考え方に適した新しいアルゴリズムの開発とその設計法を考察する。

とくに、機能が類似した能動関数の配列を導入すること、ソフトウェアが常に安全に稼働することを基本の目標に置く。

(2) アクティブソフトウェアの記述法と言語処理の方式 :

大規模で柔軟に変化するアクティブソフトウェアを開発するに当り、開発の基本となるツールは、記述のための言語とその処理系である。前項の(1)の研究の結果と、複数の能動関数が同時に起動することを考慮して、それらに適合した記述法を設定し、その記述から、逐次実行、並列実行、および、分散実行に対応できるプログラムに変換する言語プロセッサの構成方法を検討する。

(3) アクティブソフトウェアを実行するハードウェア環境 :

アクティブソフトウェアでは、能動関数の起動条件の検出方法が実行の効率に大きく影響する。また、能動関数が分散環境で移動して実行することも考慮している。

起動条件の検出を再構成可能なハードウェアで実現できるコンピュータの構成や、分散環境に合わせて移動を能動的に行なえるソフトウェアの生成の方法について調査と検討を始める。

3.3.2 平成 16 年度の研究の目標・研究計画

平成 16 年度の研究の目標

本研究では、アクティブソフトウェア(Active Software)の考えに基づいて、安全に働く新しい大規模なシステムの設計法を確立すること、および、アクティブソフトウェアを開発して実行できる環境を構築することを目的としている。

「ソフトウェアは能動的に動き出す処理の集まりである」という見方のソフトウェア設計方法の確立を目指す。

具体的には、平成 15 年度の研究目標と同じように、次の 3 つの面から実現に関する研究を進める。

(1) ソフトウェアの設計過程を分析して、アクティブソフトウェアに適した設計方法を確立する。

(2) アクティブソフトウェアの基本的な開発環境として、アクティブソフトウェアを記述する方法とそれを実現する言語プロセッサを構成する。

(3) アクティブソフトウェアを分散計算環境で実行する形態や、アクティブソフトウェアの実行に適した新しいコンピュータアーキテクチャの構成など、効率の良い実行形式を統合的に考察する。

平成 16 年度の研究実施計画

平成 15 年度には、次の研究を進めてきた。

- (1) 常に安全に稼働するアクティブソフトウェアの構成法、
- (2) アクティブソフトウェアの記述法と言語処理の方式、
- (3) アクティブソフトウェアを実行する新しいコンピュータアーキテクチャの構成の検討。

平成 16 年度はこれらを基にして、上記の研究目標に向けて、研究の進展を計る。

- (1) アクティブソフトウェアに適したアルゴリズムの開発と設計法 :

アクティブソフトウェアの構成は、計算状況を監視して、期待される計算状態に対応した処理を指示すること（何時、何に対応して、何をするか）が基本になる。

このような考え方に適した新しいアルゴリズムの開発とその設計法を考察する。

とくに、複数の能動関数が並列に動作すること、ソフトウェアが常に安全に稼働することを設計の基本目標に置く。これには、ソフトウェア自身が自分の計算状態を知って、自分の誤りを訂正したり、ソフトウェアの一部を変更したりする機能が求められる。これについても基礎的な検討を始める。

- (2) アクティブソフトウェアの記述法と言語処理の方式 :

大規模で柔軟に変化するアクティブソフトウェアを開発するに当り、開発の基本となるツールは、記述のための言語とその処理系である。

前項の研究と関連して、複数の能動関数が同時に起動することを前提とした記述法を設定し、その記述から、手続き型言語あるいはオブジェクト指向型言語で、分散環境での実行にも対応できるプログラムに変換する言語プロセッサの構成方法を検討し、その実装を進める。

- (3) アクティブソフトウェアを実行するハードウェア環境 :

アクティブソフトウェアでは、能動関数の起動条件の検出方法と並列処理の実現が実行の効率に大きく影響する。

起動条件の検出をハードウェアで並行して実現する再構成可能なコンピュータの構成について更に具体的な検討を進める。

また、能動関数がネットワークを介した分散環境で移動して実行することを考慮して、分散環境に合わせたソフトウェアの生成の方法について検討を深める。これらを統合して、新しいプログラミング環境とコンピュータアーキテクチャの開発を目指す。

3.3.3 平成 17 年度の研究の目標・研究計画

平成 17 年度の研究の目標

本研究では、「アクティブソフトウェア」の概念に基づいて、安全に働く新しいシステムの設計法を確立することと、アクティブソフトウェアの開発・実行環境を構築することを目的としている。

「ソフトウェアは、人の活動と同じように、能動的に働き出す処理の集まりである」という見方のソフトウェア設計方法の確立を目指している。

アクティブソフトウェアの設計と実現について、これまでに次の研究を進めてきた。

- (1) 能動関数を導入して、基本的な設計法を確立した。
- (2) アクティブソフトウェアの実行を追跡する機能を含めて、言語処理系を構成した。
- (3) 実行中の状況を監視して、事象の発生に適應できるコンピュータアーキテクチャの構成を考え、ハードウェアを設計して評価した。
- (4) 安全に実行することに関連して、外部からの侵入を検出できるハードウェアアーキテクチャを提案した。

これらの結果を基にして、次の局面から、実現に向けての研究を進める。

- (1) ソフトウェアの設計過程を分析して、アクティブソフトウェアに適した設計法を確立する。
- (2) アクティブソフトウェアを記述する方法と、それを実現する言語処理系を構成する。
- (3) 分散環境を含めて、アクティブソフトウェアに適した新しいコンピュータアーキテクチャの構成を考える。

これらにより、柔軟な構成と安全な実行の形式を統合的に考察する。

平成 17 年度の研究実施計画

平成 17 年度は、本研究計画の三年目（最終年度）として、次の研究を進める。

(1) 安全に働くソフトウェアの設計と記述法として、事前条件と事後条件をもった「ホーアの記法」PQR に事前チェックと事後チェックを追加することを提案した。それに基づいた設計法 ($\sim P + P$)Q(R + $\sim R$) の有効性を示すために、具体的な問題をとりあげて、新しい考え方のアルゴリズムを探求する。

できれば、その記述法を実行形式に変換する言語処理系を構成する。これによって、予測しなかった事態に対処できるソフトウェアの構成の研究を進める。

(2) アクティブソフトウェアは、「自己調整する」特徴をもつことを挙げている。これについて、プログラムの一部を変更したとき、関連する部分を整合するように変更する規則を導入した。

この考えを拡大して、一般的に、一部が変更されたり、仕様が変わったりしたときに、どのようにして自己調整するかを、形式的に考察する。(1) で採り上げた例題を通して、調整の可能性を調べる。

(3) アクティブソフトウェアに適したコンピュータアーキテクチャは、一般の広汎なイベント駆動形アーキテクチャとして、柔軟な割り込み条件を受け入れる。

走行するソフトウェアに対応して、検出すべき条件を再構成する部分もち、ハードウェアの並列処理機能を十分に活用する。(1) の記述法と関連して、例題を模擬実行して、並列性を評価する。

(4) 能動関数をベースにしたアクティブソフトウェアは、人の思考と行動のひとつの形態を模擬したものである。それに基づく設計法を考えることにより、様々な分野における「設計」の共通した概念を導き出す。

これらを統合して、新しいプログラミングパラダイムにおけるソフトウェア開発環境と、それに適合したコンピュータアーキテクチャの開発と、安全なシステムを構成する方法の確立を目指す。