5F−4

リフレクションを導入した並行オブジェクト指向言語 ABCL/c+ によるオペレーティングシステムの記述と実装*

阿隅 太志 児玉 靖司 武田 正之†
東京理科大学大学院 理工学研究科 情報科学専攻‡

1 はじめに

ハードウェア技術の進歩により、並列処理システム、分散処理システムの普及と高速化が進み、これらを有効に利用するためのオペレーティングシステムが求められている。特に、今後の計算機の有効利用のためには、ワークステーションやネットワークを用いた分散処理による計算機利用形態のみならず、携帯・移動端末をも含めた超大規模かつ動的な環境に耐え得る分散オペレーティングシステムが不可欠である。

我々は、並行オブジェクト指向言語 ABCL/c+ を用いてオペレーティングシステム Yawe を記述している。本研究では Yawe を動的な構成にすることを目的とし、ABCL/c+ にリフレクションを導入した。またそのために、並行オブジェクトの実装法を大幅に変更した。

2 並行オブジェクト指向言語 ABCL/c+

ABCL/c+ は、ABCL/1 を参考にオペレーティングシステム核を記述するための機構を導入した並行オブジェクト指向言語である。ABCL/c+ の並行オブジェクトは局所記憶、スクリプト、メッセージキュー、スレッドから構成され、休眠、活性、待機の 3 つのモードと、過去型、現在型、ルーチンオブジェクト呼出しの 3 つの通信手段を持つ。

ABCl/c+ にはオペレーティングシステム核の記述のために、通常の並行オブジェクトの他にルーチンオブジェクトを導入している。ルーチンオブジェクトは並行オブジェクトを実現するためのオブジェクトであり、スレッド、メッセージキューを持たず、またメッセージ通信はルーチンオブジェクト呼出しで行う。

*Description of operating system by reflective concurrent object oriented language ABCL/c+, and its implementation

†Taishi Asumi , Yasushi Kodama , Masayuki Takeda
‡Department of Information Sciences , Science University of Tokyo

3 ABCL/c+ へのリフレクションの導入

リフレクションの実現のためにメタオブジェクトを導入した。メタオブジェクトはその局所記憶として、ベースオブジェクトの局所記憶、スクリプト、メッセージキュー、スレッドを持つ。さらにスクリプト発火のためのディスパッチテーブルを持ち、ベースオブジェクトに到着したメッセージのキューへの格納、取り出し、解析、スクリプトの発火の制御を行う。

この他にメタオブジェクトはベースオブジェクトへのリフレクションのために、ベースオブジェクトの各部品の参照や変更のためのメッセージを受け付ける。

4 オブジェクト管理の実装

オブジェクト管理マネージャはルーチンオブジェクトで実装され、並行オブジェクトの生成、メッセージ通信、メタオブジェクトのレイファイの制御等を行う。

並行オブジェクトの管理　並行オブジェクトはオブジェクト管理マネージャが保持するスロットテーブルで管理される。オブジェクトスロットは、オブジェクトを構成する各部品やヒープ、オブジェクト状態等を所有する。

スレッド状態とオブジェクト状態は明確に区別される。並行オブジェクトはオブジェクト管理マネージャによって実現されるが、スレッドはスレッド管理ルーチンオブジェクトが実現する。オブジェクト管理マネージャは、スレッド管理マネージャに依頼することでオブジェクトスレッドの制御を行う。

メタオブジェクト　メタオブジェクトの実装は、スレッドのコンテキスト等負荷が大きいため、デフォルトのメタオブジェクトはスレッドを持たず、ベースオブジェクトを実現する上で最低限の機能だけを持つ。メタオブジェクトはリフレクション呼出し等によって並行オブジェクトとして実体化する。メタオブジェクトの実体化は同時にレイファイしていないメタメタオブジェクトが暗黙的に作成され、リフレクティブタワーが構成され

る。

今回のモデルでは、実体化されないメタオブジェクトによるリフレクティブタワーを許していない。すなわち、リフレクティブタワーはそのトップに実体化されないメタオブジェクトが存在し、その下はすべて並行オブジェクトで構成される。

レイファイ前のメタオブジェクトはベースオブジェクトのコンテキストで実行され、メタオブジェクトからベースオブジェクトのスクリプトのディスパッチは手続き呼出しで実装される。メタオブジェクトのレイファイ後は、メタオブジェクトは自身のコンテキストを持ち、ベースオブジェクトのコンテキストを操作することにより、ディスパッチを実現する。

5 リフレクティブなOSの実装

OSへのリフレクションの応用として、スケジューリングのポリシーの選択、オブジェクトのアクセス権限の変更、仮想記憶やファイルシステムのネットワーク透過、暗黙なオブジェクト移送、デバイスドライバへの適応等が考えられる。今回はその一例として、仮想記憶管理を並行オブジェクトで実装し、ポリシーの変更や仮想機械環境をリフレクティブに変更できることを確かめる。

仮想機械 OSへのリフレクションによる享受を受けるオブジェクト群を自由に定義できるようにするために、グループを定義するメタオブジェクトを導入する。このオブジェクトは、特定のグループに特有なシステムサービスを提供し、仮想機械とみなすことができる。

仮想記憶管理 仮想記憶管理オブジェクトは、その計算機のハードを直接操作するメタ仮想記憶管理オブジェクトと、各仮想機械にサービスを提供する抽象レベルのVM仮想記憶管理オブジェクトに分離して実装した。

メタ仮想記憶管理オブジェクトは、ページテーブルの切替えやページスワップ等のプリミティブな操作と、ポリシーの管理のみを行う。仮想機械毎のページテーブルの保持やポリシーの選択はVM仮想記憶管理オブジェクトが管理する。

VM仮想記憶管理オブジェクトは個々のポリシーを実現するメタオブジェクトとみなすことができる。また、ネットワーク仮想記憶や分散共有記憶はこのレベルで実装可能と思われる。ユーザレベルのオブジェクトがポリシーを変更する場合は、このVM仮想記憶管理オブジェクトにリフレクションをかける。

6 おわりに

リフレクションを導入したABCL/c+で並行オブジェクトを実装することにより、仮想記憶におけるポリシーの選択や追加、ログの生成等をリフレクティブに実現できるようになった。しかし、本研究では、グループに対するリフレクションのモデルが定まっておらず、adhocな実現となっている。今後はグループ間におけるリフレクションの定義、実装を行い、リフレクションに対する権限も設計する必要がある。

参考文献

[1] Douglas Comer. *Operating System Design - The XINU Approach.* Prentice-Hall International,Inc., 1984.

[2] Douglas Comer. *Operating System Design II - Internetworking with XINU.* Prentice-Hall International,Inc., 1987.

[3] Doi Norihisa,Yasushi Kodama. *An Implemention of an Operationg System Kernel using Concurrent Object Oriental Language ABCL/c+* in ABCL: An Object Orented Concurrent System (ed.Yonezawa,A) The MIT Press., 1990

[4] 児玉 靖司, 阿隅 太志, 土居 範久. *Yet Another Window Environment* コンピュータシステムシンポジウム, 1998.

[5] Akinori Yonezawa. *ABCL: An Object-Orented Concurrent System,* Mit Press, 1990.

[6] Satoshi Mathuoka,Takuo Watanabe,Akinori Yonezawa. *Hibrid Group Reflective Architecture for Object-Oriented Concurrent Reflective Programming.* In *Proceeding of ECOOP'91 European Conference on Object-Oriented Programming, Lecture Notes in Computer Science 512,*pp.231-250 1991.

[7] Yasuhiko Yokote. *The Apertos Reflective Operating System: The Concept and its Implementation.* In *OOPSLA '92 Proceedings,*pp.414-434.ACM, October 1992. also apeared in Sony Computer Science Laboratory Inc., Technical Report SCSL-TR-92-014.