

計算における評価と資源に基づくアーキテクチャ

5 L - 1

前川博俊、安田弘幸、沢田佳明、實藤隆則、榎本隆昭、福田謙治
ソニー(株) 総合研究所

1. はじめに

計算機アーキテクチャにおいて、データとメモリ管理やプロセス管理をあわせて資源、計算を実行する過程を評価として認識することができる。この資源と評価という観点に基づく、システムの記述性を高めることを目的とするアーキテクチャを提案する。

2. 計算の実行モデルとアーキテクチャ

関数型やオブジェクト指向などの計算モデルが与えられたとき、それを計算機上で実現するための言語やフレームワークなどの実行モデルが必要である。実行モデルは、その計算モデルによって、アーキテクチャを考慮して決められるが、実行モデルを実現するにはアーキテクチャ¹⁾上で種々のシステム記述²⁾が必要であり、一般に実行モデルとアーキテクチャとの整合性が良いとはいえない。(図1-(a))

我々が提案するアーキテクチャでは、実行モデル上の計算の基本要素をアーキテクチャモジュールとして実現することにより、実行モデルとアーキテクチャとの整合性を高める。

ばならない。

このようなアーキテクチャにおいては、実行モデルを実現したり実行モデル上での記述を行なうときに、アーキテクチャ上でのシステム記述が必ずしも簡潔ではないため、実際のシステムの動作を詳細に認識したり予測したりするのが容易ではない。その結果、メモリアクセスなどにおいて予期しないボトルネックが生じたりする。

実行モデルとアーキテクチャの整合性を高めるには、専用マシン化や機能メモリ化³⁾等の方法が考えられるが、普遍的なアーキテクチャを得るためには、多くの実行モデル上での計算の基本を与える枠組に基づく方法が求められる。我々は、実行モデル上でのデータとその計算とに着目し、データとその管理や通信などの処理を資源として独立させ、実行モデル上の計算を資源の評価として具現化するアーキテクチャ ACRE (Architecture based on Computational Resources and Evaluation) を構築した。(図1-(b))

資源とは、実行モデル上のデータや計算単位の静的表現である。資源の内部では、その管理のための処理が実行されるが、評価からそれが陽に見えることはない。

評価とは、実行モデル上での資源の動的振舞いであり、資源に外面的変化を与えるものである。その振舞いや変化は、評価のための機構上の処理によって得られる。

4. 資源と評価機構の構成

資源の領域と形態、評価の機構の詳細は実行モデルにより異なるが、その基本構成を図2に示す。

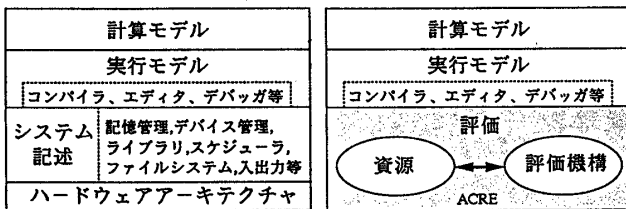


図1 計算の実行モデルとアーキテクチャ

3. 資源と評価

現在のアーキテクチャの基本をなすのは、データ及びその格納場所と、プロセッサ及びその上でのデータの演算である(フォン・ノイマン・アーキテクチャ)。実行モデルを具現化するためには、実行モデル上で示される計算の他に、メモリ管理やデバイス管理もプロセッサ上で処理しなけれ

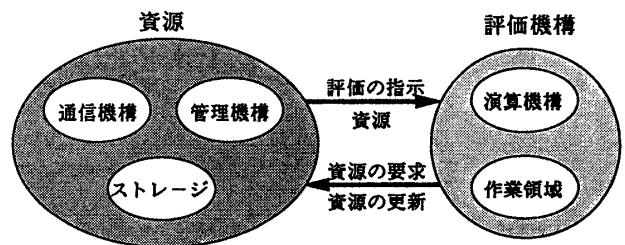


図2 資源と評価機構の構成

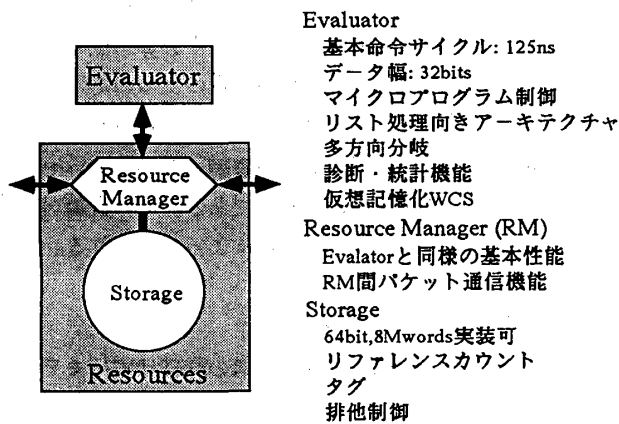
An Architecture based on Computational Resources and Evaluation
Hirotohi MAEGAWA, Hiroyuki YASUDA, Yoshiaki SAWADA,
Takanori SANETO, Takaaki ENOMOTO and George FUKUDA
Corporate Research Laboratories, Sony Corporation

資源は、データの格納場所とその管理機構や通信機構で実現する。これらの機構で処理するのは、資源内のデータとその管理によって実行モデル上のデータを表現することである。これらの処理は、仮想記憶管理やガーベジコレクションなども含む。評価を記述するとき、資源内部の処理は考慮する必要がない。

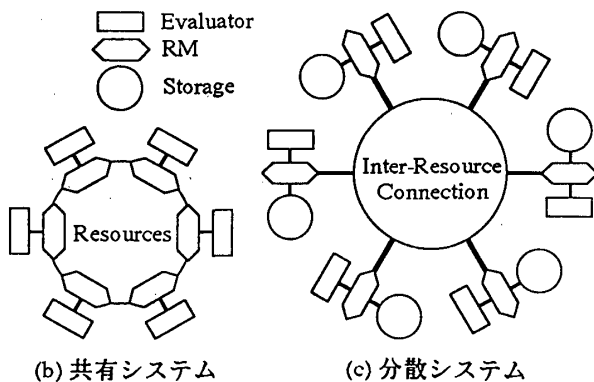
評価は、資源上のデータの演算を行なうための評価機構において実行する。評価の方法は、資源内にプログラムとして表現できる。また、その基本部分または全部を評価機構の内部に持つことも可能である。評価機構には、その演算のための作業用所データを内部に持つことができる。

5. アーキテクチャの検証

ACREを検証するため、我々は実験システム Lilac (図3)を開発中である。Lilac (Largely Integrated Language Accelerator) は、ここで述べている資源と評価という枠組に基づくアーキテクチャを持つ記号処理用システムである。RM (Resource Manager) は、従来技術のメモリを用いて資源を実現するためのエミュレータである。評価はEvaluatorで実行する。



(a) 基本構成



(b) 共有システム

(c) 分散システム

図3 Lilac システム

RMはプロセス管理やデバイス管理のほか、我々が記号処理用データのためにすでに提案している記憶構成・管理方式 SOSO (Structure Oriented Storage Organization)⁴⁻⁶⁾を実現する。

Lilacシステムでは、並列・分散処理も実現する。資源・評価機構の結合を考えた時、結合した相手も資源とみなすことができるため、Lilac では資源間の結合、即ちRMの相互結合によって並列・分散システムを実現する。(図3-(b)(c))

6. おわりに

計算の実行モデルにおける計算資源と計算過程としての評価に観点をおいて構成するアーキテクチャ ACRE を提案した。これは、実行モデルとの整合性が良く、実行モデル上の記述や実行モデルを実現するためのシステム記述をアーキテクチャに立脚して行なうことができる。

ACRE を検証するため、実験システム Lilac を開発中である。Lilac 上で、ACRE に基づくシステムの記述性、稼働性能などを、その並列・分散処理も含めて、今後確認して行く。

さらに、計算モデルとその実行モデルに応じた詳細なアーキテクチャを構築する予定である。

謝辞

日頃から御指導いただいている大阪大学 安井先生、京都大学 柴山先生、青山学院大学 井田先生に感謝します。また、本研究の機会を与えて下さった当社総合研究所 宮岡所長、総研情報通信研究所 松田所長に謝意を表します。

【参考文献】

- 1) 萩原 宏: 命令セットアーキテクチャの現状と展望, 情報処理, Vol.29, No.12, pp.1381-1386 (1988).
- 2) Donovan, J. J.: Systems Programming, McGraw-Hill (1972).
- 3) 古谷立美: 応用指向メモリ, 情報処理, Vol.27, No.6, pp.601-606 (1986).
- 4) 前川 他: Poniter-Linked Data における仮想記憶管理の一手法, 情報処理学会研究会資料, SYM50-1 (1989).
- 5) 前川 他: Linked Data Structures の記憶管理, 情報処理学会第39回全国大会 1Q-4, pp.1279-1280 (1989).
- 6) 前川 他: Linked-Data のその構造に基づく記憶空間の構成, 情報処理学会研究会資料, ARC80-5 (1990).