

Element Computing の研究*

中村 和敬

日比野 靖†

はじめに

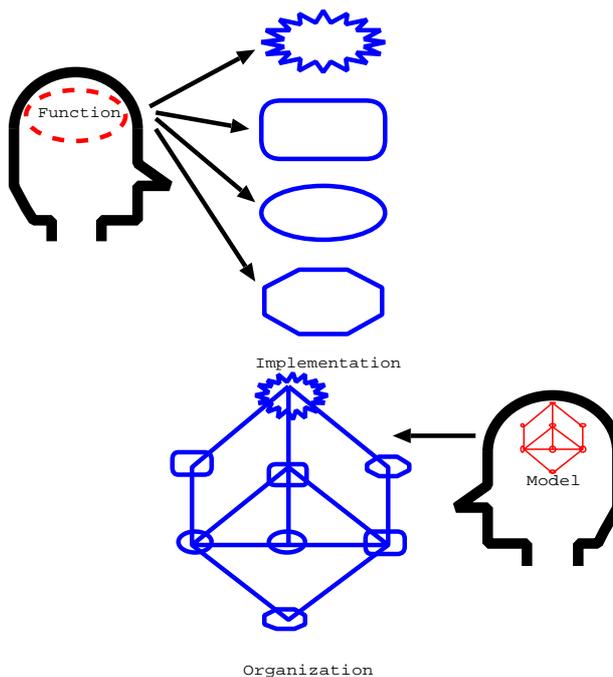
近年までに、計算機を取り巻く環境は大きく変化した。その主要な点は計算機資源の増大、広帯域ネットワークの普及、新しいデバイスの出現である。これらは複雑にからみ合い、誰もが“ユーザ”でありかつ“管理者”であるという新しい利用形態へと歩を進める事を促した。このことは新しいコンピューティング環境の出現を予感させる。

しかしながら、現状は新しい利用形態とはかけ離れておりさまざまな無駄、非効率が発生している。この理由の一つはそれを利用するためのシステムソフトウェアはかつての古いコンピュータを想定して設計された仕組みがそのまま利用されている事である。

以上の事柄をふまえ本研究において筆者らは、新しい分散システムソフトウェアが備えるべき特徴を導き出し、それによって得られるコンセプトを Element Computing と名付けた。そしてプロトタイプとして、MINIX 上のデバイスドライバをネットワーク経由で利用できる仕組みを作り上げ、Element Computing の有用性と実現可能性を検討した。

Element Computing の構成要素

Element Computing は、システム全体を相互に通信する抽象的な機能の協調動作として捉え、実装、レイヤ、ネットワークに囚われずに、機能を結合し新しい機能を作成出来るようにしようとするものである。Element Computing の考え方では、実装を隠蔽された機能を“Element”と呼び、



一つの機能に様々な実装、一つのモデルに様々な構成があるが、いずれも同じものである。

図 1: 機能と実装、モデルと構成

Element が所属する様々な通信路をまとめたものを“Abstract Network”と呼ぶ。ここで Abstract Network が Element の実装やレイヤの違いを吸収することにより、Element の機能のみに着目して新しい機能を作り出すことを可能とし、また通信路についてもよりよい方式を選択するチャンスがひろがる。さらにそうしてできあがった機能もまた Element として扱うことにより、高度な再利用性を実現できると考えられる。

*A proposal of Element Computing

†北陸先端科学技術大学院大学: JAIST

Element Computing とサービス

Element とは分散オブジェクトの単位である。Element の最小単位はハードウェアコンポーネントである。このレベルで分散環境が実現できるので、それより上位のサービスはすべて分散環境で利用できることが保証されている。そのため、システムの低位の構成を変更したいときでも、特別な操作を必要とせずに分散環境で変更を加えることができる。

ユーザはこの Element を自由に組み合わせて新しいサービスを作ることができる。Element を組み合わせて作られた新しい機能は、同様にまた Element として扱われる。このことは、ファイル、プログラム、ライブラリ、プロセス、ハードウェア、ネットワークサーバといったオブジェクトを全て Element という同じ概念で扱うということの意味する。これらのオブジェクトは実装形式の違いこそあれ、すべて何かの機能を提供するために作られた仕組みである。Element として扱うことにより、こういった実装の表面上の差異に囚われることなく各々の機能を利用することができる。

Abstract Network

Element を組み合わせて利用する際に、Element の通信路として利用されるのが、Abstract Network である。Abstract Network は Element によって拡張されることを前提としており、Abstract Network も Element の一種である。

Abstract Network の主な機能は、Element 間の通信路の違いを吸収し隠匿することである。Element 間の通信路には、レジスタ経由、メモリ経由、PCIバス、USB、インターネット、電話網、など様々な形態がある。既存のシステムでは別々の通信路を利用する処理要素同士の通信は、全く考慮されていなかった。このため、各通信路毎に処理要素が分断されてしまっていた。しかしながら、別々の通信網に属する処理要素が通信するという事は、双方の通信路の両方に属する処理要素が存在すれば、この処理要素を用いて通信することが可能である。Abstract Network はこの考

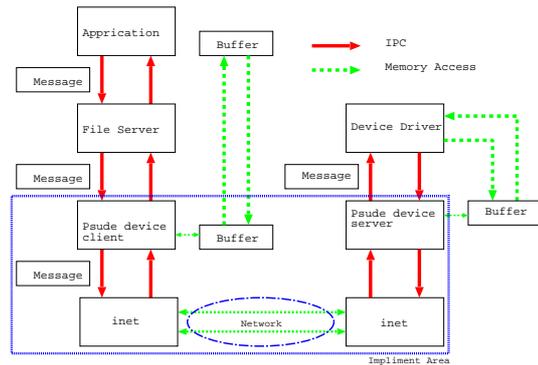


図 2: 実装

え方を用いて通信路の違いを吸収、隠匿する。これによって、Element の実装上の差違や、ネットワーク的な位置に関らず、Element 同士を結合し、新しい機能を実現することを可能にする。

また、Abstract Network は始めから Element 間のやりとりを通信として捉えるので、始めから同期の問題を扱える。これは分散システムの開発に有用な性質である。

実装

以上を踏まえて、Element Computing システムのプロトタイピングを行ない、その実現可能性を検討した。プロトタイプとしては、MINIX のデバイスドライバを、ネットワーク越しに利用する仕組みを実装した。この実装形式を選択したのは、最も基本的な Element であるハードウェアコンポーネント Element のネットワーク分散環境と比較的よく似た状態を実現できるからである。

疑似ドライバは、FS から呼び出されるクライアントと実際にドライバを呼び出すサーバに別れる。これら間でメッセージを転送し、必要ならデータを転送することによって、ドライバのリモートアクセスが実現される。各疑似ドライバはメッセージを転送する際に、MINIX システムに対して矛盾が発生しないように、メッセージを書き換える。これにより、リモートドライバの利用を可能にした。

以上により、Element Computing は実現可能であり、また有用であるとの見通しを得た。