

動的適応性を持つソフトウェアシステムの検証環境に関する検討

田中 大祐[†] 久保田 稔[†]
千葉工業大学

1. はじめに

携帯電話，組込み機器等のノード（マイクロノードと呼ぶ）が様々な場所に配置され，ネットワークで相互接続されている環境において，マイクロノードを動的に組み合わせて，新たなサービスを構成するために必要なソフトウェアシステムについて検討[1]を行っている．本稿では，実デバイス上での検証に先立ち，このシステムの動作を行うためのシミュレーション環境について述べる．

本研究の環境では，マイクロノードのシミュレーションは，仮想計算機ソフトウェア（VMware）を用いて行う．VMware を同時に複数実行させ，これらの上で，マイクロノード用のリアルタイム OS，管理用ノードの汎用 OS を実行し，これらの OS の間の通信を擬似する．

2. 想定するシステム

2.1. 動的置き換えシステム

図1に想定するシステム構成の概要を示す．マイクロノードを動的に組み合わせ，新たなサービスを構成するためには，このようなノードに搭載されるソフトウェアは極めて多様な要求条件を満足する必要がある．一方，これらの要求条件は常に変化していくことが予想される．

マイクロノードは十分なリソースを持たない物を想定しているため，あらかじめマイクロノードに必要なソフトウェアを全て作りこんでおくことは困難である．想定するシステムは，この条件下でサービスの要求条件に対応していくために，ソフトウェア自身が，環境条件やユーザーからのサービス要求条件の変化を感知し，機能を適応させていく方式が必要となる．

マイクロノードに搭載されるソフトウェアを動的に置き換えることにより，上記の問題点を解決する．

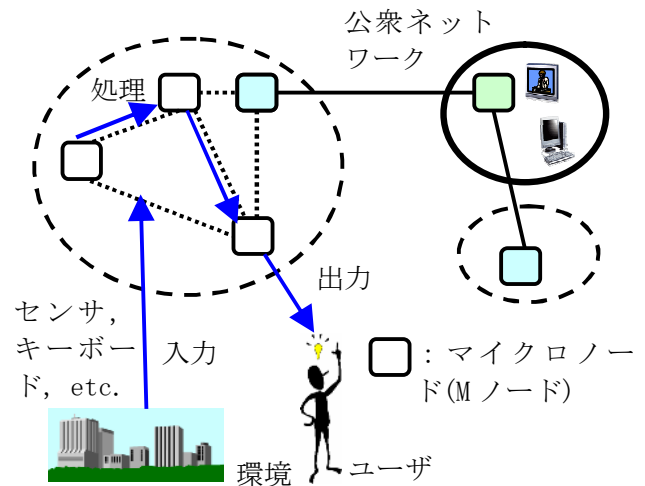


図1. システム構成の概要

2.2. システムのシミュレーション

本稿では，動的置き換えシステムの動作の検証を行うためのシミュレーション環境を構築することを検討する．図2にその概要を示す．

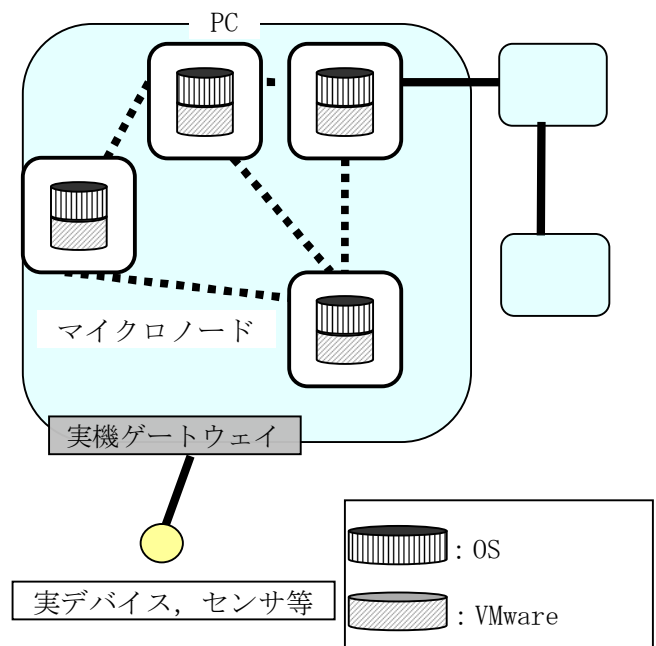


図2. シミュレーション概要

Simulation Environment for Dynamically Adaptive Software System

[†]Daisuke Tanaka [†]Minoru Kubota
Chiba Institute of Technology

実環境で検証しにくい理由として、(1)実デバイス上でプログラムをロードする場合、オーバーヘッドが大きい。(2)多数のマイクロノードを立ち上げ、検証する必要がある場合、ハードウェアの状況を監視することが困難であり、また、コストや機器の設置スペースが増大する、が挙げられる。

検証環境として、マイクロノード制御用として TOPPERS/JSP カーネル [2] をベースとしたリアルタイム OS を使用する。理由として、ターゲット独立部とターゲット依存部が分離しているため、ソースの改造、他ターゲットへの移植が容易である、GCC などのフリーの開発環境にてシステム開発が可能であること、が挙げられる。管理用ノードの汎用 OS として Windows を使用する。

VMware 上で動作している Toppers/JSP, Windows 間でパケットを送受信することにより通信を擬似する。複数のマイクロノードを VMware で擬似し、マイクロノード間で通信を行う [3] ことにより、想定する環境を実現する。

3. シミュレーション環境の構築

シミュレーション環境を構築するにあたり、課題と対策について述べる。

(1) Toppers/JSP の VMware での起動

Toppers/JSP では PC/AT 互換機のターゲットシステムがサポートされていない。Toppers/JSP を VMware で起動させるために、新規に PC/AT 互換機のターゲット依存部を作成し組み込むことにより VMware 上で動作させる。

(2) プロトコル処理機能の実装

Toppers/JSP にはプロトコル処理機能が含まれていない。

イーサネットドライバと IP パケットのみ送受信するためのプロトコル処理プログラムを作成することにより、マイクロノード間の通信を行うこととする。第 1 段階として、検証システムでは通信を確立し、データの送受信が可能であればよい。IP を用いた単純なプロトコル処理を作成することとする。第 2 段階として、TCP/IP など汎用的なプロトコル処理を移植する。第 3 段階として、無線アドホックネットワークのプロトコル処理の移植を行う。

(3) 複数マイクロノードの同時シミュレーション

マイクロノード間の通信を確立するためには、送受信する機器の判別をするための仕組みが必要である。

VMware を用いた仮想マイクロノード間を通信させるため、仮想的なネットワークを構築する。機器を判別する方法として、(a) VMware DHCP Service を用いて IP アドレスを割り当てる。(b) マイクロノードに固有の ID を付与する、がある。IP アドレスを用いて機器の判別を行ったほうが、起動したマイクロノードに対し簡単に一意の識別子を与えることができ、アドレスの管理もホスト OS から可能である。容易に仕組みを構築できるため、本稿では IP アドレスによる機器の判別を行うものとする。無線アドホックネットワーク等のも対象にした場合、IP アドレスを利用できないこともあるため、それに対応した機器判別手法が必要となる。

(4) 大規模化と実デバイスの連携

本稿で想定するシステムでは、多数のマイクロノードがいたる場所に存在することが前提にある。よって大規模な環境をシミュレーションしなければならない。複数のマイクロノードを 1 台の PC で立ち上げ、それを複数台の PC で通信することにより、大規模な環境を実現する。

また、実デバイスとシミュレーションの連携を行うことにより、より実環境に近い環境でのシミュレーションを可能にする。このため、実機ゲートウェイを設け、センサ等の実デバイスとの通信を行う。

5. まとめと今後の課題

本稿では、想定するシステム構成を実現するために、VMware 上で Toppers/JSP を用いたシミュレーション環境の構築の検証と提案に対する課題の解決方法を示した。今後、シミュレーション環境の構築を進め、ソフトウェアの動的更新技術の検証を行う。

参考文献

- [1] 久保田稔, "動的適応性をもつモジュラー型基盤ソフトウェアの提案," 信学会研技報, vol. 105, No. 670, pp. 19-24, 2006
- [2] Toppers プロジェクト, <http://www.toppers.jp/>
- [3] 中川泰宏, 須田宇宙, 浮貝雅裕, 三井田惇郎 "VMware を利用したネットワーク管理者教育の試み," 情報処理学会第 65 回全国大会 pp-4-281-4-282, 2003