

「実用化を目指した組込みシステム用ディペンダブル・オペレーティングシステム」
平成 18 年度採択研究代表者

石川 裕

東京大学 情報基盤センター 教授

並列・分散型組込みシステムのためのディペンダブルシングルシステムイメージ OS

1. 研究実施の概要

本研究領域では、ディペンダブルな情報システムの構築および運用をサポートするためのソフトウェア体系および評価指標の確立を目指している。その中で本研究課題では、並列・分散環境を前提に、以下のソフトウェア機構の研究開発を行っている。I) Linux を基に、安全かつ容易に OS 機能を拡張できる機構 P-Bus II) 時間制約保障機構および静的時間制約検証 III) CPU、メモリ、デバイスの論理分割機構 IV) クラスタリングされたコンピュータ群で単一システムイメージを提供する機構 V) 故障管理システム VI) ディペンダビリティベンチマークフレームワーク

平成 20 年度は、前年度に引き続き P-Bus をはじめとするソフトウェア機構の開発を進めるとともに、システムのディペンダビリティを評価するためのベンチマークフレームワークについての検討を進めた。

2. 研究実施内容(文中にある参照番号は 4.(1)に対応する)

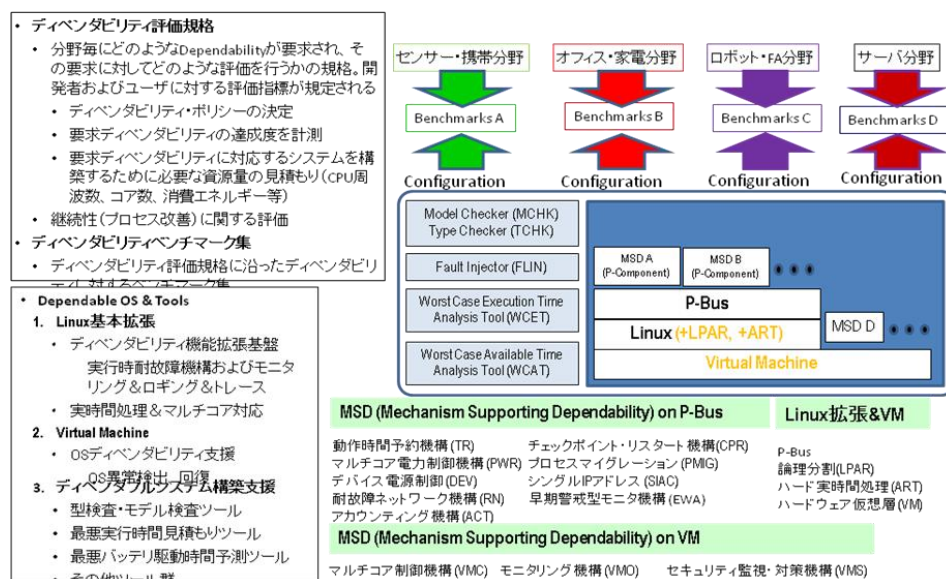


図 1 DEOS アーキテクチャ

平成 20 年度には、本研究領域全体の各研究グループの主要メンバからコアチームを構成し、コアチームにおいてシステムのディペンダビリティを実現・向上するための枠組みを検討した。平成 20 年度に検討を進めたのが図 1に示すアーキテクチャである。システムのディペンダビリティは開発時に用いられるツールと、実行時に用いられるメカニズム(MSDs)によって保証される。システムに求められるディペンダビリティの程度は当然システムの応用分野によって異なるため、分野に応じたディペンダビリティ要求が定められ、その要件を満たすためのツールとメカニズムの組み合わせ(configuration)が定められる。そして、このように構成されたシステムが実際に要求されるディペンダビリティの程度を満たしているかがディペンダビリティベンチマークによって評価される。

このアーキテクチャのもと、本課題では平成 20 年度は以下に示す研究開発を進めた。

D) P-Bus

P-Bus API の策定と開発を進めるとともに、API リファレンスマニュアルを公開した。平成 20 年度に開発を進めた P-Bus API は図 2に示す通りである。スケジューラについては平成 19 年度に開発されたプロトタイプが既に存在したが、平成 20 年度には P-Bus のメンテナンスコストの軽減を考慮し、Linux カーネルに対する動的バイナリパッチを自動で生成する Ksplice というツール(<http://www.ksplice.com>)を利用した実装への変更を行った。また、P-Bus API の使用方法に関する仕様(規則)を静的プログラム検証器に与えるという観点から、同領域でプログラム検証を担当している東大前田グループと共同で仕様記述言語の検討を行った。P-Bus のうち実装の完了した部分については研究領域内に提供し、筑波大佐藤グループの協力のもと、佐藤グループで開発されている RI2N を P-Component として P-Bus 上で動かすことに成功した。また、従来 P-Bus は x86 アーキテクチャ上で開発を進めてきたが、平成 20 年度には SH4A プラットフォームへの移植を行った。

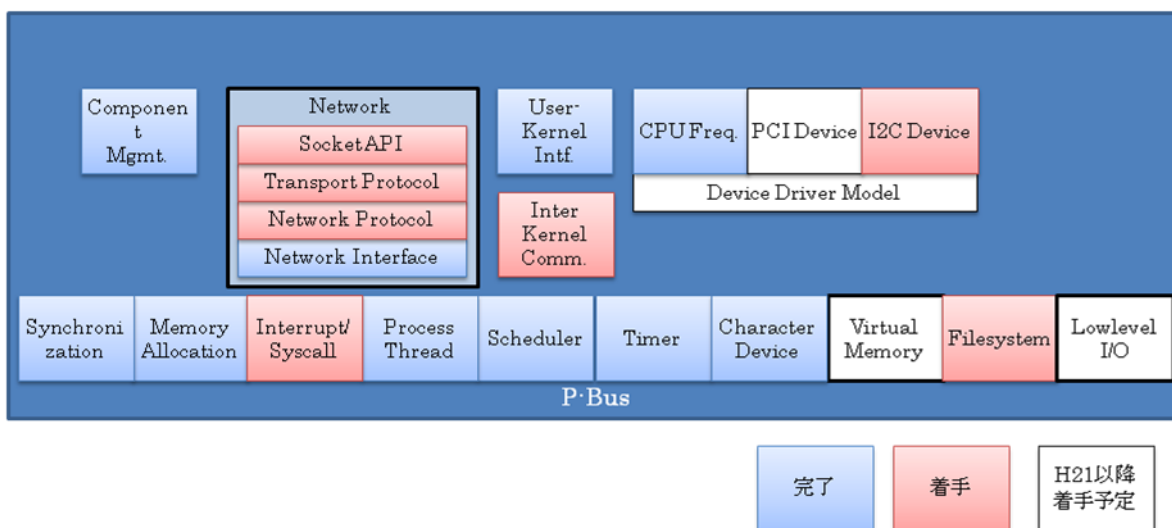


図 2 P-Bus API の分類と進捗状況

II) 時間制約保障機構および静的時間制約検証

静的 WCET(Worst Case Execution Time Analysis)検証ツールを SH4A プラットフォームに移植し、評価版リリースに向けて実装を進めた。また既存の予測手法に加えて、複数

のプロセスがスケジュールされる環境で最悪実行時間を予測する手法を検討し実装を進めた。本手法の実現には、割り込みの制御など最悪実行時間を保証するための支援機構がカーネルに必要になり、その設計を行なった。

III) CPU、メモリ、デバイスの論理分割機構

平成 19 年度に開発した機構[2]を SH4A プラットフォームに移植し、性能評価を行った。その結果、1 台の計算機を複数のカーネルで論理分割することによって、カーネル内部での資源の競合が少なくなり、1 つのカーネル上で複数の処理を同時に行うよりも、複数に分割したカーネルの上でそれぞれ処理を行った方が全体として性能が向上するということが明らかになった。また、複数カーネル間で物理デバイスを共有するために必要な機構として、カーネル間通信インタフェイスを x86 プラットフォーム上の実装に追加し、それを用いてネットワークインタフェイスおよびブロックデバイスを共有する機能を追加した。

IV) クラスタリングされたコンピュータ群で単一システムイメージを提供する機構

平成 19 年度までに開発されたプロトタイプ[1]を SH4A アーキテクチャに移植してクラスタを構成し、性能と消費電力の関係を評価した。SH4A によるクラスタを Intel Xeon プロセッサを用いたサーバマシンと比較したところ、要求される性能が低い場合には SH4A が電力効率の面で有利だが、高い性能を必要とする場合には SH4A のクラスタよりも Xeon サーバの方が有利であるということが明らかになった。単一システムイメージを実現する Linux カーネルモジュールの P-Component 化については平成 20 年度中に完了しなかったため、平成 21 年度に行う。

V) 故障管理システムフレームワーク

平成 19 年度に実装したプロトタイプをもとに、評価版リリースに向けたツール類の開発を行った。ツール類の開発によって、故障管理システムの初期化や更新が容易になった。ネットワークに関連した異常の原因を調べるプロトタイプシステムとは別に、ホスト内部の異常の発端をログに出力するシステムの設計を行った。

VI) ディペンダビリティベンチマークフレームワーク

システムのディペンダビリティを評価するためのベンチマークフレームワークを検討した。国内外の既存研究調査をしたところ、LAAS-CNRS で 1998 年から 5 年間行われた **Dependability Benchmarking** プロジェクトにおいてベンチマークフレームワークが提案されていたことが判明し、まずは、本フレームワークを基に DEOS ベンチマークフレームワークを検討した。図 3 に示す例は、シングルシステムイメージのディペンダビリティを評価するときの環境を示している。ベンチマークフレームワークの実現は平成 21 年度となる。

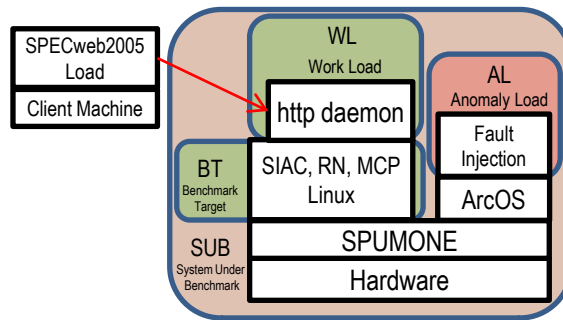


図 3 ベンチマーク例

3. 研究実施体制

(1) 石川グループ

- ① 研究分担グループ長: 石川 裕 (東京大学 教授)
- ② 研究項目

高信頼組込シングルシステムイメージ OS

4. 研究成果の発表等

(1) 論文発表 (原著論文)

1. Hajime Fujita, Hiroya Matsuba, Yutaka Ishikawa, "TCP Connection Scheduler in Single IP Cluster", 8th IEEE International Symposium on Cluster Computing and the Grid (CCGRID'08), pp. 366-375, May 2008
2. Taku Shimosawa, Hiroya Matsuba, Yutaka Ishikawa, "Logical Partitioning without Architectural Supports", 32nd IEEE International Computer Software and Applications Conference (COMPSAC 2008), pp. 355-364, 2008

(2) 特許出願

平成 20 年度 国内特許出願件数 : 0 件 (CREST 研究期間累積件数 : 0 件)