「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

平成 20 年度
実績報告

中島　達夫

早稲田大学　理工学術院　教授

高機能情報家電のためのディペンダブルオペレーティングシステム

1. 研究実施の概要

本研究チームでは、仮想化を利用した情報家電機器向けのディペンダブル OS の構築に取り組む。本研究が目指す、ディペンダブル OS における仮想化とは、本来の OS が提供するメカニズムを大きく変更することなく、システム全体のディペンダビリティの向上を可能とする。仮想化により、システム内で発生する障害の効果の低減、障害発生時の再起動の制御、マルチコアプロセッサにおけるリソース管理の強化を可能とする。

本年度は、マルチコアプロセッサを仮想化する仮想マシンのプロトタイプの構築、モニタリングシステムの構築、Linux のロギングシステムの構築をおこなった。また、これらのシステムを DEOS ソフトウェアアーキテクチャに統合するための議論をおこなった。

2. 研究実施内容（文中にある参照番号は 4.(1)に対応する）

本研究チームでは、以下に示す２つの面から仮想化を利用した情報家電機器向けのディペンダブル OS の構築に取り組む。

１つ目の仮想化技術はモニタリングサービスを利用した Linux カーネル内のエラーの仮想化である。Linux カーネルはソフトウエアのバグや外部からの悪意のある攻撃により、カーネル内のデータ構造の一貫性が失われる可能性がある。DEOS では、P-Bus が提供する検証機能により型非安全なエラーが発生しないことが保証されているが、型安全なエラーへの対応は P-Bus では不十分である。本研究が提供するエラーの仮想化機能では、データ構造の一貫性制約ルールを定義し、その一貫性が失われたときに、エラーからの修復をおこなうことにより、データ構造の一貫性が保たれることを保証する。それにより、仮想的にエラーの発生が起きなかったように見せることを可能とする。ここでは、データ構造の一貫性修復のことをエラーの仮想化と呼ぶ。データ構造の一貫性修復は、データ構造を完全にロールバックすることを前提としていない。前方回復と後方回復の両方を利用可能とすることにより、回復可能性を向上することを目指している。Linux にはガーベジコレクション機能がないため、前方回復を利用した場合は、一部のデータを解放できないケース

も存在する。例えば、rootkit による攻撃の場合は、最終的に rootkit を消去するためには、システムを再起動する必要があるケースも存在する。そのため、本研究では、下に述べるハードウエア抽象化層と連動させることにより、システム全体の一貫性を回復するために定期的な再起動を利用する。

２つ目の仮想化技術は、プロセッサを仮想化するためのハードウエア抽象化層の提供である。本研究では、プロセッサ仮想化は２つの役割を担当する。１つ目は、モニタリングサービスを実行するための基盤環境の提供である。モニタリングサービスのインテグリティを保証するため、モニタリングサービスは Linux の外部で実行する。P-Bus が型非安全なエラーや攻撃による障害が発生しないことを保証するので、Linux カーネル外のメモリへのアクセスはソフトウエア的に禁止される。そのため、プロセッサ仮想化は原則として OS 間の隔離機能を提供する必要がないので、リアルタイム性をそこなうことなく性能が高い仮想化機能の実現が可能となる。２つ目は、マルチコアプロセッサへの対応である。本研究が提供するプロセッサの仮想化は、システム全体の負荷に応じて、適切な数のコアの利用を可能とする。特に、情報家電機器は、使用するときと使用しない時の負荷が大きく変化する。そのため、システムの負荷が低い場合は、起動するプロセッサコアの数を最小限にすることにより、消費電力量を出来るだけ少なくすることが好ましい。本研究により開発するプロセッサ仮想化は、ゲスト OS が使用するプロセッサを仮想プロセッサとして扱うことを可能とすることにより、物理的に利用するコア数を動的に変化することを可能とする。

我々が構築する仮想化技術は、大きく４つのコンポーネントから構成される。1 つ目のコンポーネントはハードウェア抽象化、2 つ目は、モニタリングサービス、3 つ目は、高信頼 RTOS、4 つ目はロギングサービスモジュールである。

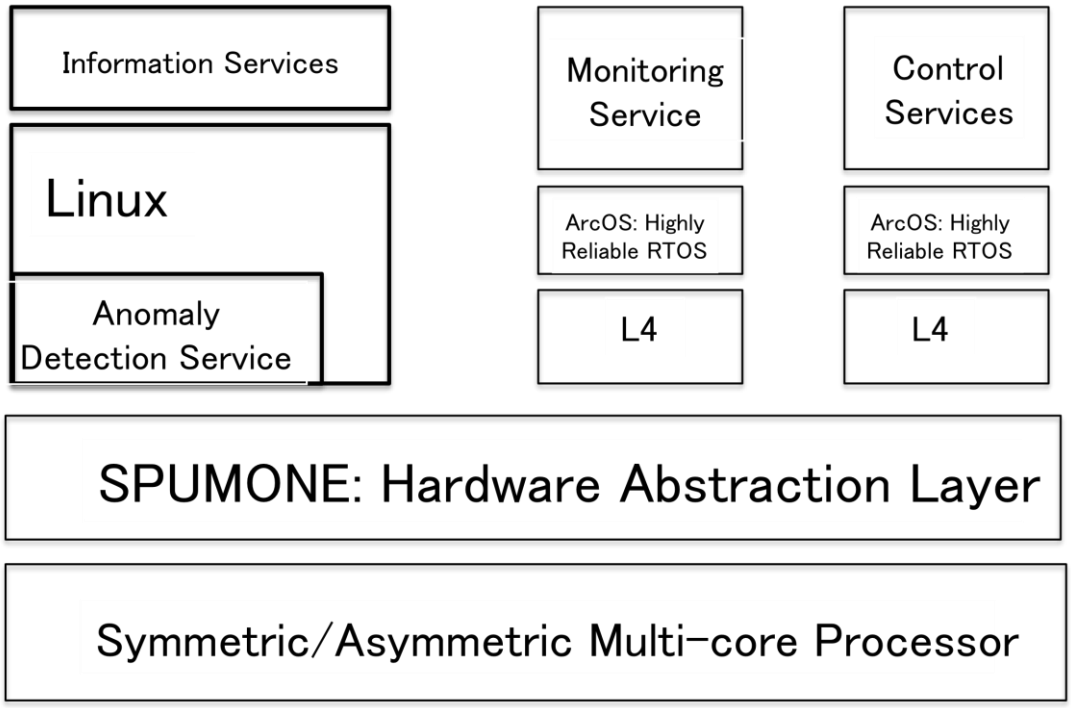

図１　中島チームが開発するシステムの全体アーキテクチャ

ハードウェア抽象化層は、モニタリングサービスを Linux カーネルからアイソレーションするための機能を提供する。ハードウェア抽象化層はシングルコアプロセッサとマルチコアプロセッサの両方で動作する。本研究で利用する Linux カーネルはハードウェア抽象化層上で動作するように変更する必要がある。モニタリングサービスは Linux カーネル内に異常が発生したことをモニタリングする。モニタリングサービス自体は信頼性を向上するため、それ自体が再起動可能なフレームワーク上に実装される。そのため、モニタリングサービス自体が障害クラッシュした場合、自分自身を即座に再起動することにより高い信頼性を保証する。高信頼 RTOS は、実時間アプリケーションやマルチコア向け並列アプリケーションを実行するための基盤となる OS である。本 OS はマイクロカーネル型の OS として実現され、ファイルシステムやデバイスドライバ等の OS コンポーネント自体を容易に再起動することが可能となる。ロギングサービスモジュールは、動的にメモリを割り当てるプロセスが高負荷時にメモリを確保できずにクラッシュしないようにメモリを予約したり、実時間プロセスが必要とする CPU キャパシティを予約することを可能とする。また、使用するリソースキャパシティ情報を利用することにより、システムの異常を発見したり、システムの性能向上を可能としたりすることを可能とする。

本年度は、第一の仮想化をおこなうためのモニタリングシステムのプロトタイプシステムの実現とハードウエア抽象化層のプロトタイプシステムの構築をおこない、その結果を査読付きの国際会議において９通の論文発表をおこない（ハードウエア抽象化層に関して 3 件[1-1, 1-4, 1-9]、モニタリングシステムに関して 4 件[1-2, 1-3, 1-5, 1-8]、高信頼 RTOS に関して１件[1-6]、ロギングシステムに関して 1 件[1-7])、また、１通の論文誌が採択された[1-9]（印刷中)。

3. 研究実施体制

(1)「早稲田大学」グループ

①研究分担グループ長:中島　達夫(早稲田大学、教授)

②研究項目　仮想化を利用したディペンダブルＯＳの構築

(2)「追川」グループ

①研究分担グループ長:追川　修一(筑波大学、准教授)

②研究項目　ディペンダブルOSの x86 対応

4. 研究成果の発表等

（1）論文発表（原著論文）

1-1) Yuki Kinebuchi, Midori Sugaya, Shuichi Oikawa, Tatsuo Nakajima, ``Task Grain Scheduling for Hypervisor-Based Embedded System", In Proceeding of the 10th IEEE International Conference on High Performance Computing and Communications (HPCC-08), pp. 190-197, Sept. 2008.

1-2) Sun Lei, Tatsuo Nakjaima, ``A Lightweight Kernel Objects Monitoring Infrastructure for Embedded Systems", In Proceedings of The 14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA 2008). pp. 55-60, Aug, 2008.

1-3) Lei Sun, Tatsuo Nakajima, ``A Lightweight Detection and Recovery Infrastructure of Kernel Objects for Embedded Systems", In Proceedings of The 2008 International Conference On Embedded and Ubiquitous Computing(EUC 2008), pp. 136-143 , Dec, 2008.

1-4) Wataru Kanda, Yuki Kinebuchi, Yu Yumura, Tatsuo Nakajima, ``SPUMONE: LightWeight CPU Virtualization Layer for Embedded Systems", In Proceedings of The 2008 International Conference On Embedded and Ubiquitous Computing(EUC 2008), pp. 144-151, Dec, 2008.

1-5) Tomohiro Katori, Lei Sun, Dennis Nilsson, Tatsuo Nakajima, ``Building a Self-Healing Embedded System in a Multi-OS Environment", In the proceedings of the 24th Annual ACM Symposium on Applied Computing, March, 2008.

1-6) Hiroo Ishikawa, Alexandre Courbot, Tatsuo Nakajima, ``A Framework for Self-healing Device Drivers", Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems, pp. 277-286, Oct, 2008.

1-7) Midori Sugaya, Yuki Ohno, Andrej van der Zee and Tatsuo Nakajima, "A Lightweight Anomaly Detection System for Information Appliances" , 12th IEEE International Symposium on Object/component/service-oriented Real-time distributed Computing (ISORC 2009) Tokyo, Japan, March 17-20, 2009.

1-8) Lei Sun, Dennis K. Nilsson, Tomohiro Katori and Tatsuo Nakajima, "Online Self-Healing Support for Embedded Systems" , 12th IEEE International Symposium on Object/component/service-oriented Real-time distributed Computing (ISORC 2009) Tokyo, Japan, March 17-20, 2009.

1-9) Wataru Kanda, Yu Murata and Tatsuo Nakajima, SIGMA System: A Multi-OS Environment for Embedded Systems, Journal of Signal Processing Systems, In Press, 2009.

（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）