

[研究論文]

省電力サーバシステムと高可用性サーバシステムの複合動作による省電力かつ高可用性サーバシステムの開発

Development of a Power-saving and High-availability Server System by Compound Operation of a Power-saving Server System and a High-availability Server System

北村 光芳[†], 田中 龍馬[‡]

Mitsuyoshi KITAMURA, Ryoma TANAKA

[†] 東京工芸大学大学院 工学研究科

[‡] 株式会社日立情報通信エンジニアリング

[†] Graduate School of Engineering, Tokyo Polytechnic University

[‡] Hitachi Information & Telecommunication Engineering, Ltd.

要旨

単独で動作可能な省電力サーバシステム (Power-saving Server System: PSS) と高可用性サーバシステム (Ring-type Dynamic Backup Server System: RDBSS) を複合動作させることで、専用の管理サーバを必要としない省電力かつ高可用性サーバシステムを開発する。PSS と RDBSS の特徴、複合動作を実現する場合の問題点及びその解決方法を示す。また、PSS 及び RDBSS の管理プログラムにおける動作効率の悪化を防ぐために、システムインタフェース方式を採用する。本提案方式を採用した実験システムを構築し、サーバシステムの構成変更やサーバ故障を再現する実験を行い、PSS と RDBSS の単独及び複合動作時における処理時間の比較を行う。実験結果から、提案方式による省電力かつ高可用性サーバシステムが安定稼働することを示す。

Abstract

A power-saving, high-availability server system not requiring a dedicated management server is developed by the compound operation of a power-saving server system (PSS) and a high-availability server system (Ring-type Dynamic Backup Server System: RDBSS), both of which can operate independently. Characteristics of the PSS and the RDBSS, problems involved in realizing compound operations, and solutions to these problems are described. Moreover, a system interface method is adopted to prevent a deterioration in the operating efficiency of the PSS and RDBSS management programs. An experimental system based on the proposed method was constructed and experiments to reproduce server system configuration changes and server failures were conducted. In the experiments, we compared the processing times for independent and compound execution of the PSS and the RDBSS. The experimental results demonstrate that the power-saving, high-availability server system developed using the proposed method operates stably.

1. はじめに

近年、インターネットサービスは我々の生活に深く浸透している。そのため、そのサービスをより便利にするためにデータセンタの市場規模も拡大している[1]。これらのサービスを提供するサーバシステムの最適化や安全稼働が非常に重要となる。

IT システムの障害は、社会経済活動や組織の信頼に甚大な影響を及ぼす。2021 年以降、国内大手銀行においてデータベースの不具合やハードウェアの故障に起因するシステム障害が頻発し、ATM の稼働停止、振込処理の遅延、窓口業務の停滞を招いた。また、2021 年の AWS (Amazon Web Services) 東京リージョンにおける大規模障害では、ネットワーク機器の故障により、証券取引、EC サイト、物流システム等の多岐にわたるサービスの中断が発生した。そのため、IT システムの高可用性に関する検討の重要性が再認識されている。

また、世界的なエネルギー情勢の不安定化や円安の進行で、電気料金が高止まりしており、データセンタやサーバールームを運用する企業にとって、直接的な利益圧迫要因となっている。加えて、2050 年のカーボンニュートラル実現[2]に向け、企業は温室効果ガス (CO₂) 排出量の削減を求められている。IT 機器の稼働に伴う電力消費は、企業活動における CO₂ 排出量の相当な割合を占める事例が多く、その削減は非常に重要となる。これに対するグリーン IT[3]のアプローチとして、IT システムの消費電力を削減する取り組みがあり、具体的には自社サーバなどのクラウドへの移行やサーバ仮想化技術の活用など

[研究論文]

2025 年 9 月 25 日受付, 2026 年 2 月 27 日改訂, 2026 年 3 月 18 日受理

© 情報システム学会

が挙げられる。

したがって、エネルギー効率最適化と高可用性を両立させるサーバ管理システムの確立は、IT 基盤における重要課題となる。そこで、本研究では、中小規模のオンプレミス環境やエッジコンピューティング環境での使用を想定し、単独で動作可能な専用の管理サーバを必要としない省電力サーバシステム (Power-saving Server System: PSS) [4]と動的バックアップサーバシステム (Dynamic Backup Server System: DBSS) [5]をリング型に配置する専用の管理サーバを必要としない高可用性サーバシステム (Ring-type Dynamic Backup Server System: RDBSS) [6]を複合動作させることで、専用の管理サーバを必要としない省電力かつ高可用性サーバシステムの開発を目的とする。ここで、本論文で使用する RDBSS ([3.2 高可用性サーバシステム (RDBSS) の特徴]を参照) は、Microsoft 社の Redirected Drive Buffer Subsystem (RDBSS と略されることが多い) とは異なる。各サーバシステムの特徴を示し、複合動作を実現する場合の問題点を詳細に示すとともに、その解決方法を示す。また、PSS 及び RDBSS における管理プログラムの動作効率を悪化せずに複合動作を実現するため、システムインタフェース方式 (System Interface: SIF) [7]を専用の管理サーバを必要としないサーバ管理方式に適用し、採用する。本提案方式を採用した実験システムを構築し、PSS と RDBSS の単独及び複合動作時において、サーバシステムの構成変更及びサーバ故障時における復旧時間を実測し、検討を行う。実験結果から、PSS と RDBSS の複合動作による省電力かつ高可用性サーバシステムが安定稼働することを示す。

本論文の残りの部分は以下のように構成する。2. では実装システム及び関連研究について示し、3. では PSS と RDBSS の特徴及びシステム全体の構成概要を示す。4. では複合動作時における問題点とその解決方法及び SIF の詳細について示す。5. では実験システムの構成や仕様を示すとともに実験結果及び考察を述べ、最後に結論を 6. で述べる。

2. 実装システム及び関連研究

近年では持続可能なサーバシステムの構築や運用コスト削減が重要視されている。そのため、省電力化に関する研究が検討されている。広帯域密度光送受信モジュールを採用し、省エネルギーかつ高性能なサーバシステムの提案[8]や映像配信システムにおいて、ネットワーク処理とストレージアクセス処理を専用のハードウェアで行うことで省電力化を実現する手法が報告されている[9]。サーバシステムにおいて、エネルギー消費量の削減は重要な課題となる。しかし、文献[8]ではシリコンフォトリソグラフィを利用したモジュールや広帯域幅を持つ FPGA カードの導入が必要となり、文献[9]では専用のハードウェアを使用するため、コストや拡張性が懸念される。また、IoT アプリケーションにおいて、サーバスリープ・ウェイク制御を最適化し、省電力化と情報のリアルタイム性維持を両立させる研究[10]やエッジサーバにおいて QoS の最適化を維持しながらタスクを処理するアクティブサーバ数を減らすことで省電力化を実現する管理手法[11]が提案されている。しかし、実システム上での検証がされていない。

サーバシステムの設計において、システム障害からの迅速な復旧は非常に重要な要素となる。高可用性サーバシステムの実装システムとして、HA クラスタシステム (High Availability Cluster System: HACS) [12]やロードバランサ (LB) による負荷分散クラスタシステム (Load Balancing Cluster System: LBCS) [12]の採用例が多い。HACS では、冗長化対象のサーバ群が同一サービスとなる構成が基本となるため、様々なサービスを提供する場合を想定すると管理の複雑化が懸念される。そのため、複数種類のサービスを提供する高可用性サーバシステムでは LBCS の採用例が多い。しかし、サーバ故障により冗長化対象のサーバ数が減少する問題を有している。

クラウド技術において、AWS Auto Scaling[13]や Kubernetes の Horizontal Pod Autoscaler (HPA) [14]は、需要に応じたリソースの自動増減を実現するマネージドサービスとして広く普及している。これらはリソースの集約率を高めることで、データセンタ全体のエネルギー効率を間接的に改善している。しかし、本提案システムは、サーバ負荷の低下時やクライアントからのアクセス減少時に応じて実サーバ自体をサスペンドすることで、直接的な消費電力削減を実現している。また、従来の Auto Scaling や HPA には、省電力を優先すれば応答性能が低下し、可用性を優先すれば過剰な待機電力が発生するという課題を有している。これに対し、本システムは省電力制御と高可用性維持の 2 系統を複合的に動作させることから、そのような課題は存在しない。

高可用性に関する研究として、IoT アプリケーションにおいて、分散エッジコンピューティングを利用した低遅延タスク管理方式と高可用性を実現するシステムの提案[15]やクラウド環境において、リソースなどに基づき障害を事前に予測、対処する FT-ERM フレームワークが報告[16]されている。文献[15]では一定数のバックアップサーバの確保により高可用性を実現しているが、複数のバックアップサーバを必要とし、文献[16]では複数機器を用いて監視と予測を行う。そのため、ともに導入及び運用コスト

が懸念される。また、高可用性分散クラスタにおいて、復旧時間を短縮するために、サーバやネットワークの異常を予測し、データの冗長性を動的に制御するシステムが提案[17]されている。しかし、ネットワーク異常検知技術と連携させた実環境での実装、動作検証については実施されていない。更に、高可用性サーバシステムの構築においてコストの低減を考慮し、1つの管理プログラムが1台のサーバのみを管理するDBSSを1台の管理サーバ上に複数動作させ、複数台のサーバ管理を行う複合サーババックアップシステム（Multiple Server Backup System: MSBS）[5]が報告されている。しかし、一般的なサーバ管理システムと同様で専用の管理サーバが監視や管理を行うため、その故障時が懸念される（サーバ管理システムにおける単一障害点）。

そこで、専用の管理サーバを必要としない省電力サーバシステム（PSS）[4]やDBSSをリング型に配置する専用の管理サーバを必要としないRDBSS[6]が提案されている。専用の管理サーバを必要としないサーバ管理方式はサーバ管理システムの単一障害点である管理サーバの故障に対して非常に有効である。しかし、文献[4]は省電力化に重点を置き、サーバ故障時の対策としては、簡易的な死活監視を採用しているため、高可用性に関する能力が懸念される。また、文献[6]は高可用性に関して有効であるが、省電力に関する検討がされていない。

省電力かつ高可用性サーバシステムの開発に関する研究も進められている。1台の管理サーバ上にMSBSと省電力サーバシステムを複合動作することで、省電力かつ高可用性サーバシステムを構築する研究[7]が進められている。ここでは1台の管理サーバ上にSIFを採用し、MSBSと省電力サーバシステムにおける管理プログラムの動作効率を悪化せずに複合動作を実現している。サーバシステムにおいて、省電力かつ高可用性の実現は非常に重要である。しかし、専用の管理サーバが監視や管理を行うため、その故障時が懸念される。また、管理プログラムの構成を考慮した省電力かつ高可用性サーバシステムの構築方法が報告[18]されている。しかし、実験による検証がされていない。ここで、既存研究[4][6][7][18]を基礎として本提案システムを構築していることから、表1に上述した既存研究との違いをまとめる。比較項目としては「専用の管理サーバが不要」、「省電力化」、「高可用性」、「稼働プログラムの種類」、「稼働実験による動作検証」とし、“○”が対応可、“×”が対応不可、“△”が懸念事項ありを示す。

表1 本提案システムと既存研究との比較

	Proposed system	Reference 4	Reference 6	Reference 7	Reference 18
No required dedicated management server	○	○	○	×	○
Power saving	○	○	×	○	○
High availability	○	△	○	○	○
Operating program types	3	1	1	3	3
Operation verification through operational experiments	○	○	○	○	×

サーバ管理システムの開発において、「導入コストを考慮すること」、「省電力かつ高可用性の機能を有していること」、「管理プログラムの構成を考慮すること」、「管理サーバなどのサーバ故障に対応していること」や「実験によりその動作確認ができていること」が重要であるが、上述した実装システムや関連研究では、このすべてを兼ね備えたシステムは存在しない。そこで、本論文の新規性は、以下の点をすべて同時に実現していることにある。

- ・専用の管理サーバを必要としないサーバ管理方式：
サービス提供サーバ自体が管理機能を実行することで、コスト削減と単一障害点の排除を両立している。
- ・省電力と高可用性の同時実現：
両機能を独立した系統として複合動作を行う。
- ・保守性を考慮したプログラム構成：

省電力、高可用性及びそれらを制御する SIF へとプログラムを分割し、サーバ管理者の負担を軽減できる。

・実環境での動作検証：

SIF を専用の管理サーバを必要としないサーバ管理方式に適用し、実験によりその有効性を実証している。

3. 省電力かつ高可用性サーバシステム

3.1. 省電力サーバシステム (PSS) の特徴

図 1 に導入対象である PSS[4]の特徴を示す。実サーバ RS1～RS3 が仮想サーバを起動している。仮想サーバは Mail (M1, M2), Web (W1, W2), Login (L1, L2) のサービスを提供し、それぞれのサーバはサービスグループを構成している。Ms, Ws と Ls は冗長化構成を維持するためのスペアサーバとする。Base server はサーバシステムの省電力化構成において、稼働状態を維持する実サーバを示しており、図 1(b)では Base server1 と 2, 図 1(c)では Base server1 のみが起動状態となる。このシステムでは、専用の管理サーバを必要としないサーバ管理方式のため、各仮想サーバに PSS の管理プログラムが起動しており、管理サーバかつ管理対象サーバとなる。各サービスグループの負荷やアクセス数を定期的の実測し、最適なサーバシステム構成を実現する。負荷は、CPU の使用率、メモリの使用量とネットワークの転送量を基に算出している。ここで、PSS では省電力化に重点を置いており、サーバの高可用性に関しては、個々のサーバが内部でのみ調査を行う簡易的な死活監視を採用し、異常を検出した場合、サーバ機能の復旧処理を行う。そのため、高可用性に重点を置く RDBSS との複合動作を考慮し、この一連の監視及び復旧処理は廃止する。

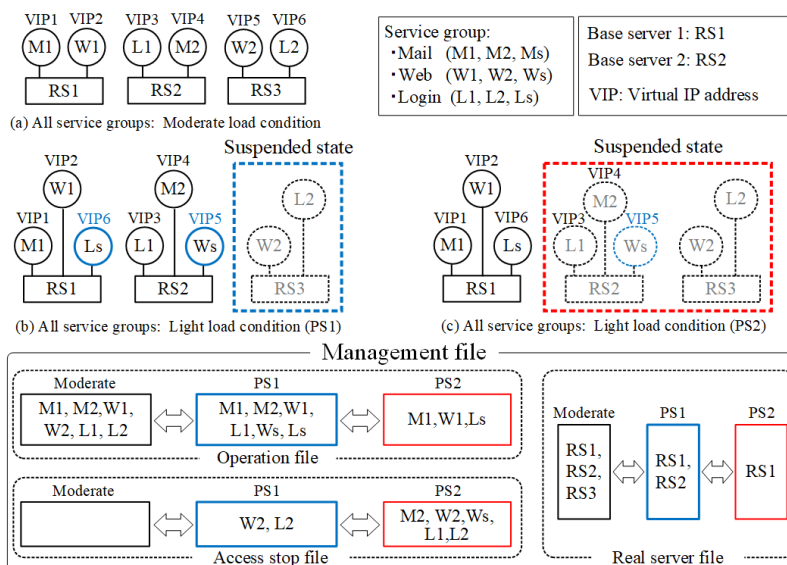


図 1 PSS の特徴

通常の稼働状態を Moderate load condition とし、図 1(a)に示すように、実サーバは 3 台稼働し、サービスグループ毎の仮想サーバは冗長化構成を構築している。クライアントからのアクセスは LB 経由で、仮想 IP アドレス (VIP) に対して行われる。各サービスグループにおける負荷のしきい値や連続回数は、過去のシステム稼働状態を考慮し、設定する。定期的な負荷の実測を行い、しきい値より低いと判断される状態が設定した回数分連続した場合、図 1(b)に示す Light load condition (PS1)に移行する。ここでは、RS3 をサスペンド状態に制御し、実サーバ RS1 (Base server1) と RS2 (Base server2) のみが稼働状態となることで、省電力化を実現する。仮想サーバの冗長化構成を維持するために、スペアサーバ Ws と Ls の起動及び VIP の設定を行う。RS3 上に起動されている W2 と L2 は LB からアクセスがされない状態 (アクセスストップ状態) となる。クライアントから各サービスグループへのアクセス数のしきい値や連続回数は、過去のシステム稼働状態を考慮し、設定する。定期的なアクセス数の実測を行い、サーバの負荷値と同様にしきい値より低いと判断される状態が設定した回数分連続した場合、図 1(c)に示す Light load condition (PS2)に移行する。ここでは、実サーバ RS2 をサスペンド状態に制御し、RS1 (Base

server1) のみが稼働状態となることで、より省電力化の効率を高めている。稼働サーバが RS1 のみとなるため、仮想サーバは非冗長化構成となる。

管理ファイルとして、PSS 用 Operation ファイルには稼働している仮想サーバの情報が記録され、Access stop ファイルにはアクセスストップ状態である仮想サーバの情報が記録される。また、Real server ファイルには稼働している実サーバの情報が記録される。図 1 下段部分の管理ファイルは、図 1(a)~(c)の状態における稼働状態の遷移を示している。稼働状態が Moderate load condition から Light load condition (PS1)に移行する場合、各管理ファイルは Moderate から PS1 の状態に、Light load condition (PS1)から Light load condition (PS2)に移行する場合には、各管理ファイルは PS1 から PS2 の状態に編集される。また、Light load condition (PS2)から Light load condition (PS1)に戻る場合、各管理ファイルは PS2 から PS1 の状態に、Light load condition (PS1)から Moderate load condition に戻る場合には、PS1 から Moderate の状態に戻される。

3.2. 高可用性サーバシステム (RDBSS) の特徴

図 2 に導入対象である RDBSS[6]の概要を示す。ここでは、LB を用いて構成される仮想サーバシステムに RDBSS を導入する。実サーバ、仮想サーバ及び仮想 IP アドレスの表記は図 1 と同様とする。RDBSS は専用の管理サーバを必要としないサーバ管理方式のため、各仮想サーバは管理プログラムとして DBSS を起動し、管理サーバかつ管理対象サーバとなる。LB が各仮想サーバのネットワーク及びサービス提供状態の監視を行うため、ここでの RDBSS はサーバ同士が監視（ネットワーク及びサービス提供状態）を行う機能を廃止する。各仮想サーバの DBSS は、管理データ（管理対象サーバの IP アドレス、ホスト名、IP アドレスや提供サービスのデーモン名などが記録）を引数として実行し、LB からの故障情報が送られるまで待機状態となる。

図 2 に示すように LB が仮想サーバ W1 の故障を検出した場合（図 2 の上段部分）、その故障情報を各仮想サーバに送信する。各サーバは管理構成を示す Operation ファイル（図 2 の下段部分）を確認する。故障直後では Operation ファイルの左の状態となるため、M1 が故障サーバ W1 の復旧処理及び各種管理ファイルの更新を行う。M1 は各実サーバの負荷及び仮想サーバの起動台数を調査し、それらを総合した値が一番低い RS2 に W1 と同様のサービスを提供可能なバックアップサーバ Wb の起動制御を行う。起動を確認後、W1 に設定していた VIP2 を Wb に設定し、W1 の機能を復旧する。ここで、W1 が復旧した場合、元の構成に戻り、W1 が復旧しない場合、Wb を管理構成に追加する。

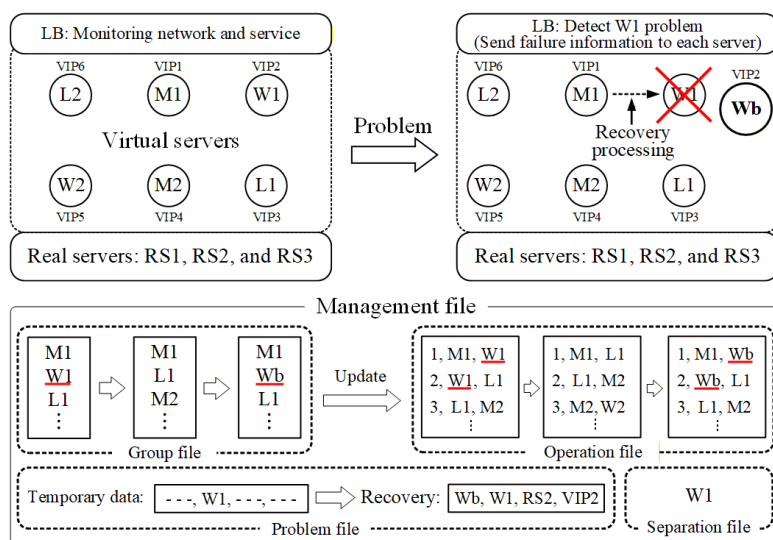


図 2 RDBSS の概要

管理ファイルとして、仮想サーバの情報は Group ファイルに記録され、それを基に管理構成を示す Operation ファイルが作成される。図 2 に示すように、システム起動時には Group ファイルの左に示す状態となり、作成される Operation ファイルも左に示す状態となる。W1 が故障した場合、W1 の情報は Group ファイルから Separation ファイルに移行される。そのため、Group ファイルは中央に示す状態に編集され、作成される Operation ファイルも中央に示す状態となる。W1 が復旧しない場合、Group ファイルに Wb を追加するため、Group ファイルは右に示す状態に編集され、作成される Operation ファイルも右に

示す状態となる。故障した W1 の復旧状況は Problem ファイルに記録される。故障直後は一時的に故障サーバ名 W1 のみが記録され、復旧処理が確定（RS2 上に Wb を起動し、VIP2 を設定）した場合、図 2 に示す Recovery の状態となる。

3.3. 省電力かつ高可用性サーバシステムの概要

図 3 に省電力かつ高可用性サーバシステムの概要を示す。図 3 では、実サーバ RS1～RS3 が仮想サーバ群を起動している。各仮想サーバは、Mail, Web, Login のサービスグループを構成しており、各グループの仮想サーバは冗長化構成を構築している。クライアントからのアクセスは LB 経由で各サーバに接続される。各仮想サーバはクライアントにサービスを提供しており、管理サーバの機能も有している。ここでは、PSS と RDBSS を複合動作させることで、専用の管理サーバを必要としない省電力かつ高可用性サーバシステムを構築している。

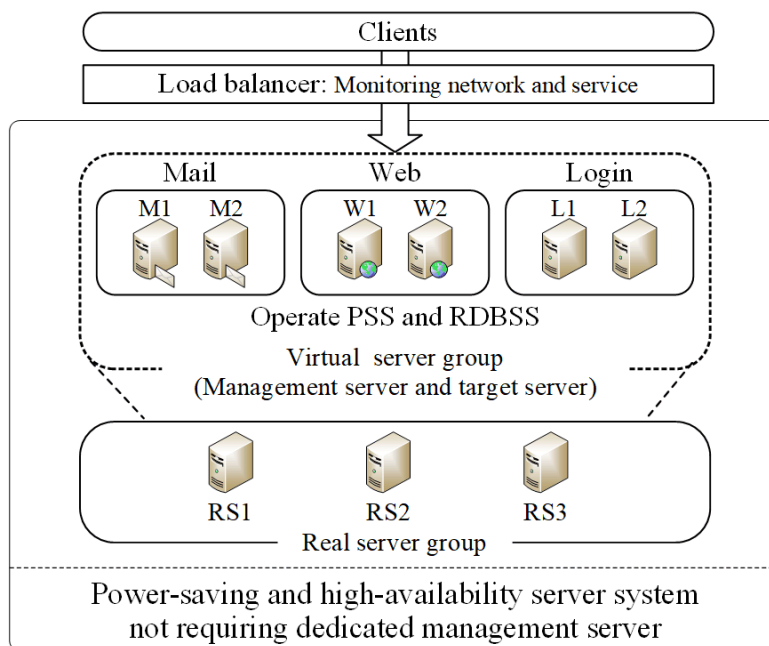


図 3 省電力かつ高可用性サーバシステムの概要

4. 複合動作時における問題及びその解決方法

4.1. 仮想サーバ故障時における問題とその解決方法

クライアントにサービスを提供する仮想サーバが故障または復旧した場合の PSS における問題点及びその解決方法を図 4 に示す。PSS は、PSS 用 Operation ファイルに記録されている仮想サーバによって構成される各サービスグループの負荷測定を行う。仮想サーバが故障した場合、PSS はその仮想サーバが所属するサービスグループにおいて正常な負荷測定ができなくなる。図 4(a)に示すように M2 が故障した場合、DBSS (M2) (M2 の管理を行う DBSS) がバックアップサーバ Mb の起動制御を行い、M2 のサーバ機能を復旧する。この状態では、PSS は Mb の存在が分からないため、サービスグループ Mail の正常な負荷測定ができない。この問題を解決するためには、DBSS (M2) が PSS 用 Operation ファイルを編集 (M2 のデータ行を削除、Mb のデータ行を追加) し、Mb 上に PSS 用管理プログラムを起動する必要がある。

また、M2 が復旧した場合、同様の問題が発生する。この問題を解決するためには、図 4(b)に示すように Mb から M2 に戻すため、DBSS (M2) が PSS 用 Operation ファイルを編集 (Mb のデータ行を削除、M2 のデータ行を追加) し、Mb 上で動作している PSS 用管理プログラムの停止及び M2 上に PSS 用管理プログラムを起動する必要がある。

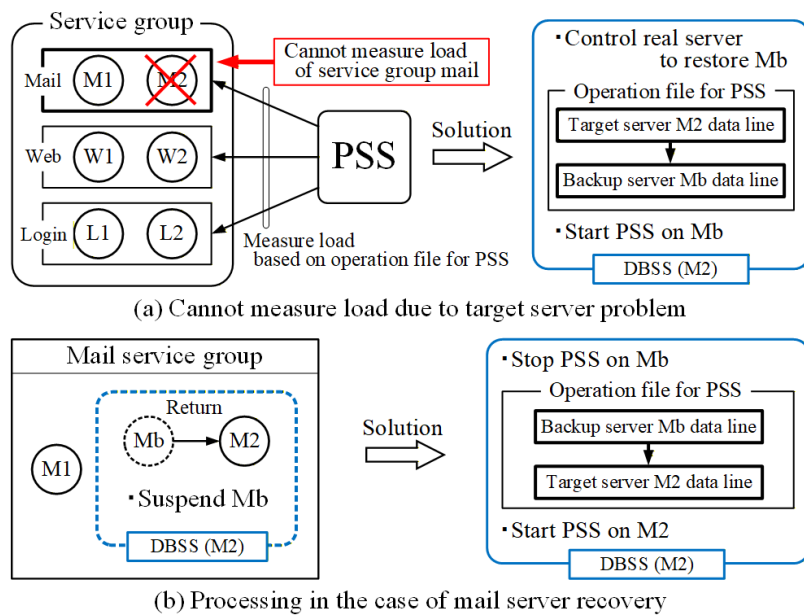


図4 サーバ故障や復旧時の問題点及びその解決方法

4.2. サーバシステムの構成変更による問題の分析

図5にPSSによるサーバシステムの構成変更時におけるRDBSSの必要処理を示す。

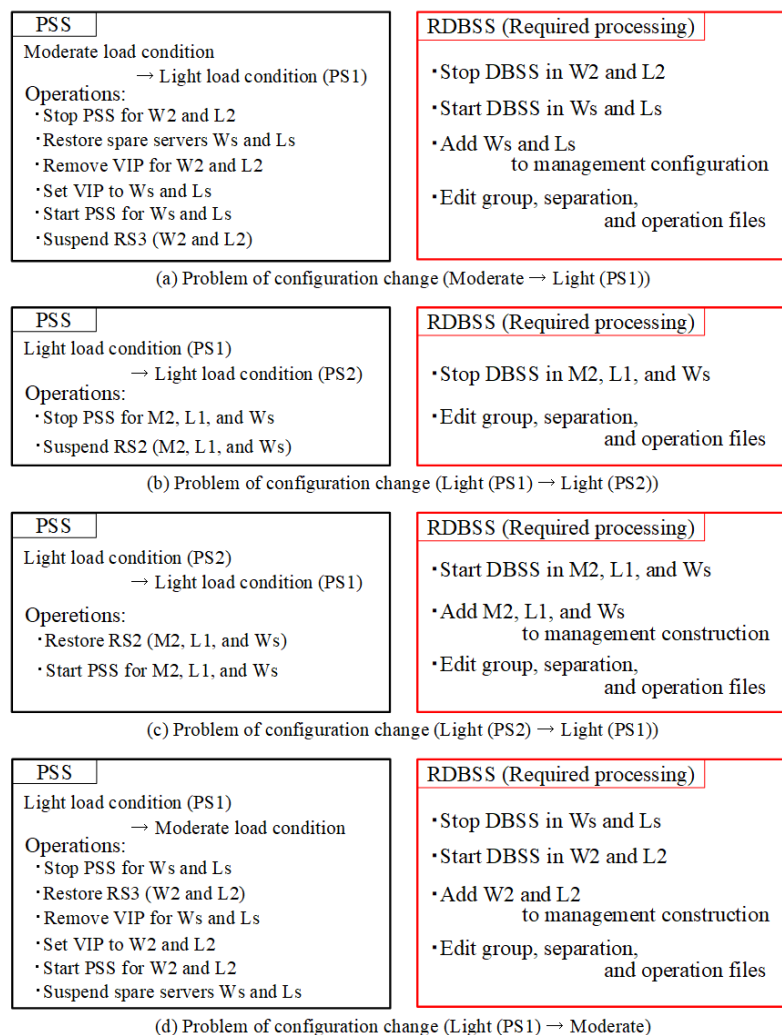


図5 サーバシステム構成変更時におけるRDBSSの必要処理

Moderate load condition から Light load condition (PS1) に構成変更する場合、図 5(a) の左 (PSS) に示すように、PSS は W2 と L2 の PSS 用管理プログラムの停止、スペアサーバ Ws と Ls の起動、VIP の削除及び追加設定、Ws と Ls の PSS 用管理プログラムの起動や実サーバ RS3 をサスペンド状態 (W2 と L2 はアクセスストップ状態) に制御する。そのため、図 5(a) の右 (RDBSS) に示すように、RDBSS は W2 と L2 における DBSS の停止、Ws と Ls における DBSS の起動、Ws と Ls を RDBSS の管理構成に追加や変更後の構成に応じた各管理ファイルの編集を行う必要がある。

Light load condition (PS1) から Light load condition (PS2) に構成変更する場合、図 5(b) の左 (PSS) に示すように、PSS は M2, L1 と Ws の PSS 用管理プログラムの停止や RS2 をサスペンド状態 (M2, L1 と Ws はアクセスストップ状態) に制御する。そのため、図 5(b) の右 (RDBSS) に示すように、RDBSS は M2, L1 と Ws における DBSS の停止や変更後の構成に応じた各管理ファイルの編集を行う必要がある。

Light load condition (PS2) から Light load condition (PS1) に構成を戻す場合、図 5(c) の左 (PSS) に示すように、PSS は実サーバ RS2 を起動 (M2, Ws と L1 はサービス提供状態) と M2, L1 と Ws の PSS 用管理プログラムの起動を行う。そのため、図 5(c) の右 (RDBSS) に示すように、RDBSS は M2, L1 と Ws における DBSS の起動及び管理構成への追加や変更後の構成に応じた各管理ファイルの編集を行う必要がある。

Light load condition (PS1) から Moderate load condition に構成を戻す場合、図 5(d) の左 (PSS) に示すように、PSS は Ws と Ls の PSS 用管理プログラムの停止、実サーバ RS3 の起動 (W2 と L2 はサービス提供状態)、VIP の削除及び追加設定、W2 と L2 の PSS 用管理プログラムの起動や Ws と Ls をサスペンド状態に制御する。そのため、図 5(d) の右 (RDBSS) に示すように、RDBSS は Ws と Ls における DBSS の停止、W2 と L2 における DBSS の起動、W2 と L2 を RDBSS の管理構成に追加や変更後の構成に応じた各管理ファイルの編集を行う必要がある。

しかしながら、3.2 で示したように各 DBSS は LB からの故障情報が送られるまで待機状態となっている。そのため、上記の必要処理を正常なタイミングで実行できない。

4.3. サーバシステム構成変更時の誤動作防止

RDBSS との複合動作を実行するため、PSS 単体動作時の処理フローに 4.2 で示した追加処理 (A1～A4) を図 6 に示す。この図では Moderate load condition から Light load condition (PS1) 及び Light load condition (PS1) から Light load condition (PS2) に移行する場合の処理フローを示す。

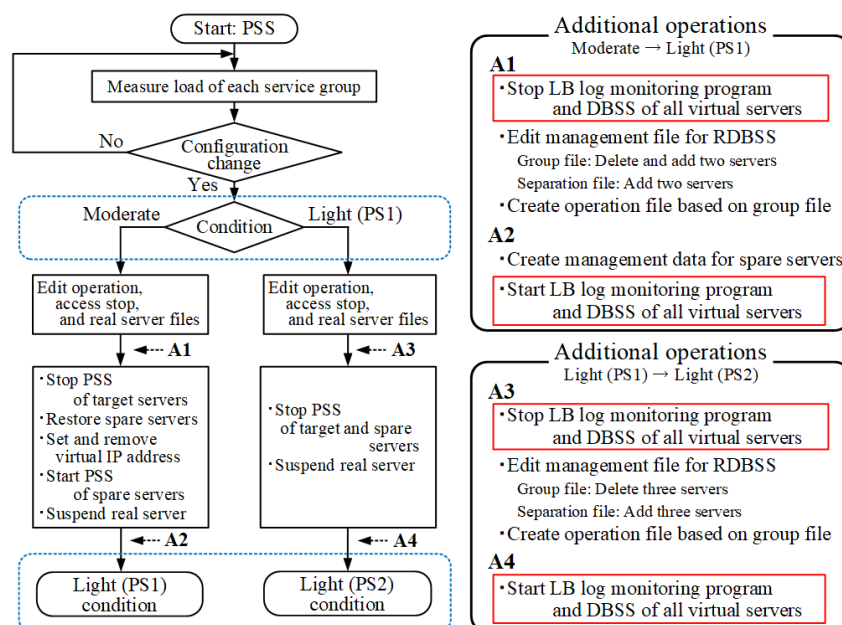


図 6 構成変更時の誤動作防止

Moderate load condition から Light load condition (PS1) に移行する場合において、追加処理 A1 では、構成変更を行う前に RDBSS の停止処理 (LB の Log 監視プログラム及び各サーバ上の DBSS の停止) を行い、RDBSS 用の管理ファイルを Light load condition (PS1) の構成用に編集する。PSS による構成変更終了後、追加処理 A2 ではスペアサーバ Ws と Ls の RDBSS 用管理データを作成し、RDBSS の開始処理 (LB

の Log 監視プログラムの起動及び各サーバ上の DBSS の起動)を行う. Light load condition (PS1)から Light load condition (PS2)に移行する場合において, 追加処理 A3 では, RDBSS の停止処理を行い, RDBSS 用の管理ファイルを Light load condition (PS2)の構成用に編集する. 追加処理 A4 では RDBSS の開始処理を行う.

4.4. SIF による複合動作方法

4.1 及び 4.3 に複合動作する上で必要となる処理を述べた. この処理を PSS の管理プログラムや DBSS に導入すると, 以下の理由から, その動作効率の悪化やプログラムの複雑化が懸念される.

① プログラム行数の増加:

PSS 及び DBSS 双方の複数箇所に相互制御用のプログラム (管理ファイルの更新機能等) を追加する必要があり, ソースコードが肥大化する.

② 条件分岐の複雑化:

単独動作と複合動作を切り替えるための条件文が随所に必要となる.

③ 保守性の低下:

改変やバグ対処の場合, 点在する複合動作の処理を常に考慮しなければならず, 作業負担が増大する.

④ DBSS の待機時間発生:

サーバ故障発生時, DBSS が PSS への通知処理を完了するまで自身の処理が待機状態となる.

⑤ PSS の待機時間発生:

サーバシステムの構成変更時, PSS が RDBSS への通知処理を完了するまで自身の処理が待機状態となる.

また, 全サーバが管理機能を担う本システムでは, 全サーバでこれらの複雑なプログラムを稼働させる必要があり, 管理コストの更なる増大が予想される.

そこで, SIF[7]を専用の管理サーバを必要としないサーバ管理方式に適用するために, 全サーバに SIF を導入する. 図 7 に SIF の概要及びそれへの指示例について示す.

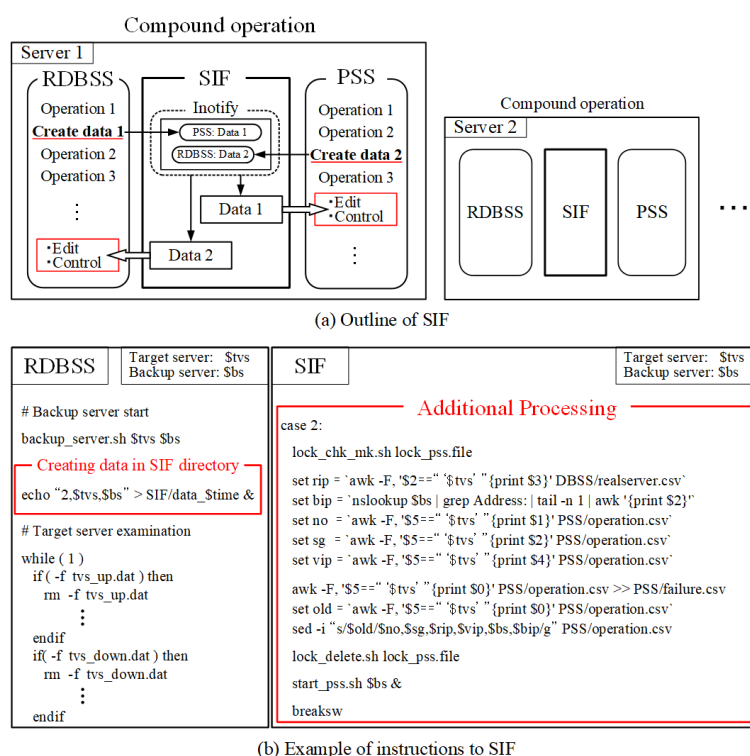


図 7 SIF の概要及び指示例

本提案システムは図 7(a)に示すように, RDBSS, PSS と SIF が 1 台のサーバ上で同時に稼働しており, この構成が全サーバ上で動作している. RDBSS や PSS は, SIF へプロセス間通信によって指示を与える. 具体的には, SIF はデータレシーブ用のディレクトリ (メモリディスク上) にデータが作成されると,

その内容に応じて、RDBSS または PSS の制御や管理ファイルの編集を行う。データレシーブ用ディレクトリを Linux の Inotify により監視しており、データが作成されると、割り込み処理によって指示された内容の処理を行うため、高速な対応を実現している。

ここで、図 7(a)の Server1 において、RDBSS が PSS に対する処理用の Data1 を SIF のデータレシーブ用ディレクトリに作成後、直ちに次ステップ (Operation2) を実行できる。これと並行して、SIF が Inotify によって Data1 を検出し、PSS に対する処理を非同期に実行する。PSS から RDBSS への指示 (Data2) についても同様である。

図 7(b)に RDBSS が PSS の管理ファイルを編集する場合の指示例 (C シェルスクリプト) を示す。SIF を適用することで、前述の課題①～⑤は以下のように解決される。

①, ②の解決:

RDBSS 側への追加コードは指示データを作成する 1 行 (図 7(b)の左) のみで済み、大幅に簡素化される (図 7(b)右の例では、SIF を用いない場合には条件文を含み 13 行の増加となる)。また、Inotify の監視停止のみで単独動作へ切り替えられるため、複雑な条件文も不要となる。

③の解決:

複合動作に関する制御ロジックは SIF に集約されるため、保守や変更が容易になる。

④, ⑤の解決:

上述したように、指示データの作成直後に自系処理を続行できるため、相互の処理完了を待機することによる動作効率の悪化を回避できる。

以上のように、SIF の導入により、各管理プログラムの動作効率を維持したまま、省電力と高可用性の複合動作が可能となる。

図 8 に PSS と RDBSS の SIF による複合動作方法の詳細を示す。SIF のデータレシーブ用ディレクトリに作成するデータはシグナル番号とサーバ名で構成される。ここでは、サーバが故障した場合と Moderate load condition から Light load condition (PS1)に移行する場合の SIF による処理を示す。

M2 が故障した場合、RDBSS ではバックアップサーバ Mb を起動し、M2 のサーバ機能を復旧する。M2 を管理する DBSS はこの情報を PSS に伝えるため、シグナル 1 とサーバ名 (M2, Mb) が記入されたデータを SIF のデータレシーブ用ディレクトリに作成する。SIF はそのデータに基づき PSS 用 Operation ファイルを編集 (M2 の情報を Mb の情報に変更) し、Mb 上に PSS 用管理プログラムを実行する。

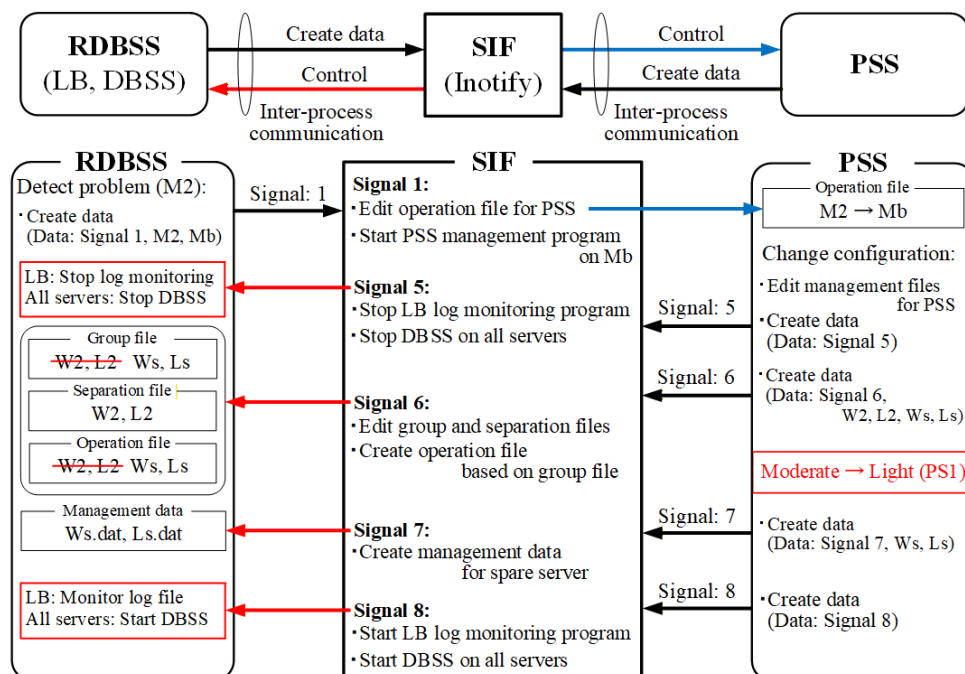


図 8 SIF による複合動作方法の詳細

Moderate load condition から Light load condition (PS1)に移行する場合は、PSS の管理プログラムは管理ファイルを編集後、シグナル 5 が記入されたデータを SIF のデータレシーブ用ディレクトリに作成する。SIF はそのデータに基づき RDBSS (LB における Log 監視及び各サーバの DBSS) の停止処理を行う。

次に、シグナル 6 が記入されたデータを SIF のデータレシーブ用ディレクトリに作成すると、SIF はそのデータに基づき RDBSS 用の管理ファイル等の編集を行う。PSS による構成変更終了後、シグナル 7 が記入されたデータを SIF のデータレシーブ用ディレクトリに作成する。SIF はそのデータに基づきスペアサーバの RDBSS 用管理データを作成し、各サーバに送信する。次に、シグナル 8 が記入されたデータを SIF のデータレシーブ用ディレクトリに作成すると、SIF はそのデータに基づき RDBSS (LB における Log 監視及び各サーバの DBSS) の開始処理を行う。

5. 動作実験

本実験では、各サーバに Linux の stress コマンドを使用し、CPU 及びメモリ負荷を与えて Moderate load condition を作成する。その stress コマンドを停止することで Light load condition (PS1)への構成変更を促す。また、Light load condition (PS1)から Light load condition (PS2)への構成変更はしきい値を変更することで実行する。更に、サーバ故障の再現において、サーバストラブルはサービスを提供するデーモンの停止、ネットワークトラブルはサーバをシャットダウンすることで再現する。そのため、ここではサーバシステムの構成変更及びサーバ故障時における復旧処理に関し、提案システムの挙動を調査する範囲とし、実測する各時間は動作の有効性を確認するための参考値として示す。

5.1. 実験システムの構成及びその仕様

図 9 に示すように実験システムは、3 台の実サーバ (RS1, RS2, RS3) が 6 台の仮想サーバを起動している。各仮想サーバはクライアントに Mail, Web, Login のサービスを提供し、それぞれのサービスグループに所属する仮想サーバは冗長化構成を構築している。Mail サービスグループには、M1 と M2, Web サービスグループには、W1 と W2, Login サービスグループには、L1 と L2 が所属し、各仮想サーバでは PSS, RDBSS 及び SIF が動作している。ファイルサーバはそれぞれのサービスグループにおけるサービスの冗長化用及び PSS や RDBSS の管理ファイルを共有するために使用し、各仮想サーバと NFS 接続をしている。クライアントにサービスを提供するネットワークにはスイッチング HUB1, ファイルサーバへのアクセス用にはスイッチング HUB2 を使用する。図 9 において 1000BASE-T のネットワークで接続される部分を実線で示し、RS1 から起動する仮想サーバ (M1, W1) を破線、RS2 から起動する仮想サーバ (M2, L1) を点線、RS3 から起動する仮想サーバ (W2, L2) を一点鎖線で示す。

故障したサーバのサーバ機能を復旧するためのバックアップサーバ及び構成変更時に冗長化構成を維持するためのスペアサーバは各実サーバ上に仮想サーバ用システムディスクを設定する。本システムは自動複製オリジナルサーバ方式[19]を採用しており、必要数分のバックアップサーバを自動的に追加可能としている。

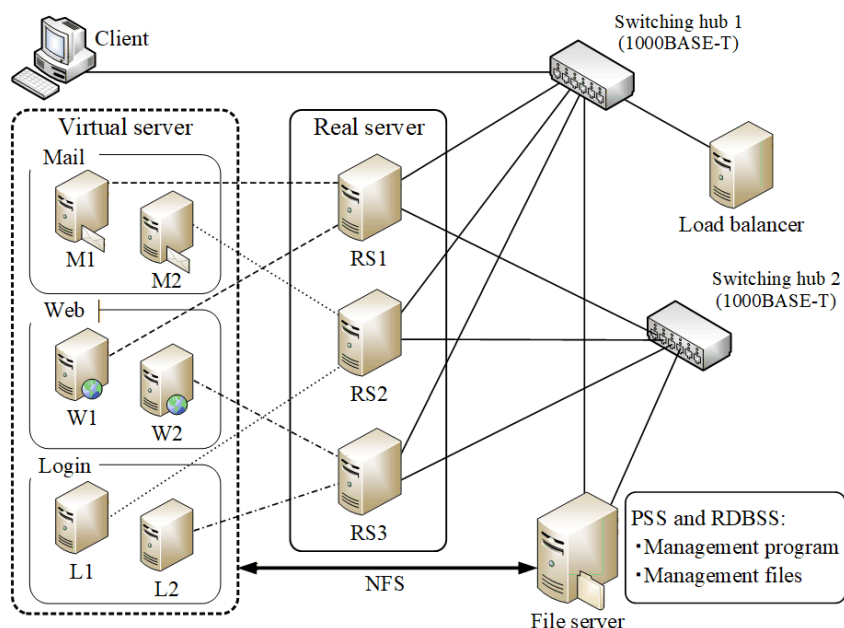


図 9 実験システムの構成

表 2 に実験システムの仕様、特性及び設定値を示す。仮想サーバを複数台起動することから実サーバの物理メモリは 8,192MB、ストレージに SSD を採用する。サーバ仮想化技術としては、Kernel-based Virtual Machine (KVM) を採用している。LB にはソフトウェア処理方式を採用し、HAProxy をインストールする。また、実サーバ、仮想サーバ及び LB にはサーバ用 OS の Rocky Linux (Red Hat Enterprise Linux のクローン OS) をインストールし、メモリを有効活用するために CUI 環境で動作させる。実サーバの起動時間を T_{rs} 、サスペンド状態からの起動時間を T_{ra} とする。また、仮想サーバ、スベアサーバやバックアップサーバの通常起動時間を T_{vsn} 、シャットダウン直後の状態からの起動時間を T_{vsc} (ディスクキャッシュの影響で T_{vsn} より約 4 秒短い)、再起動時間を T_{vr} 、サスペンド状態からの起動時間を T_{va} 、仮想 IP アドレスの設定時間を T_{vv} とする。ここで、 T_{vsn} は 18.25 秒で、 T_{va} は 2.05 秒と非常に高速となる。そのため、バックアップサーバやスベアサーバは常時、サスペンド状態で待機させる。設定パラメータとして、ネットワークの切断を判断する時間 (T_n) を 2.00 秒、サービスの停止を判断する時間 (T_s) を 3.00 秒、仮想サーバが再起動中かどうか判断するための時間 (T_b : T_{vr} の約 1.5 倍) を 22.00 秒とする。

表 2 実験システムの仕様、特性及び設定値

Specifications	Real server (RS1, RS2, RS3)	File server	Characteristics	Real server	
			Start time (s)	T_{rs}	29.41
CPU	Intel Core i7-4790 3.60 GHz (TB: 4.00 GHz)		Time to activate suspended server (s)	T_{ra}	4.42
Memory	8,192 MB		Characteristics	Virtual server/ Spare server/ Backup server	
Storage	SSD (SATA 3)		Start time (s) (No cache / Cache)	$T_{vsn}/$ T_{vsc}	18.25/14.17
Software	KVM 9.5.0	nfsd	Restart time (s)	T_{vr}	15.88
OS	Rocky Linux 9.5		Time to activate suspended server (s)	T_{va}	2.05
Specifications	Virtual server/ Spare server/ Backup server		Virtual IP address setup time (s)	T_{vv}	1.13
CPU	1 Core		Setting parameters		
Memory	2,048 MB		Network examination time (Disconnection) (s)	T_n	2.00
Software	postfix, Apache, sshd		Maximum waiting time for service examination (s)	T_s	3.00
OS	Rocky Linux 9.5		Maximum waiting time for server restart (s)	T_b	22.00
Specifications	Load balancer				
CPU	Intel Core i5-4430 3.00 GHz (TB: 3.20 GHz)				
Memory	4,096 MB				
Software	HAProxy 2.4				
OS	Rocky Linux 9.5				

5.2. 実験結果及び分析

PSS 及び RDBSS において、単独及び複合動作時での実験をそれぞれ 10 回行い、処理時間及びその変動係数を示す。電力の測定にはシステムアートウェア社のワットアワーメータ SHW3A を使用する。PSS における構成変更に必要な時間や消費電力を表 3 に示す。Difference は複合動作時と単独動作時における処理時間の差を示す。

Moderate load condition から Light load condition (PS1)への構成変更に必要な時間では、サスペンド状態からの起動時間 (T_{va}) と仮想 IP アドレスの設定時間 (T_{vv}) が主な要素となり、約 3.5 秒となっている。Light load condition (PS1)から Light load condition (PS2)への構成変更に必要な時間では、構成変更前後の RDBSS の停止や開始処理など、非常に短時間で終了する処理の加算により、約 1.4 秒となっている。Light load condition (PS2)から Light load condition (PS1)への構成変更に必要な時間では、実サーバのサスペンド状態からの起動時間 (T_{ra}) が主な要素となり、約 4.6 秒となっている。Light load condition (PS1)から Moderate load condition への構成変更に必要な時間では、実サーバのサスペンド状態からの起動時間 (T_{ra}) と仮想 IP アドレスの設定時間 (T_{vv}) が主な要素となり、約 6.0 秒となっている。4 種類の構成変更における変動係数の結果から各構成変更時の処理は安定しており、それぞれの Difference の値には差が殆

ど無いことが分かる。また、PSS による構成変更の前後において、RDBSS の誤動作などは見られなかった。サーバシステムの消費電力は Moderate load condition を基準とし、Light load condition (PS1)では約 71%、Light load condition (PS2)では約 39%となった。

表 3 PSS における構成変更や消費電力の特性

Required time for configuration change (s) (Coefficient of variation)				Load condition	Power consumption (W)
Load condition	Compound operation	Single operation	Difference		
Moderate → Light (PS1)	3.609 (0.050)	3.503 (0.022)	0.106	Moderate	95.523
Light (PS1) → Light (PS2)	1.430 (0.046)	1.355 (0.043)	0.075	Light (PS1)	68.270
Light (PS2) → Light (PS1)	4.610 (0.001)	4.584 (0.005)	0.026	Light (PS2)	37.390
Light (PS1) → Moderate	6.020 (0.001)	5.934 (0.000)	0.086		

1 台または 2 台同時にサービス及びネットワークトラブルを再現した場合の RDBSS におけるサーバ機能の復旧時間及び管理対象サーバの復旧時間を表 4 に示す。DBSS が故障を検出 (LB から故障情報を取得) 後から、バックアップサーバに VIP を設定するまでをサーバ機能の復旧時間とし、故障サーバが復旧するまでを管理対象サーバの復旧時間とする。DBSS において複合動作に追加した SIF へのデータ作成プログラムはサーバ機能の復旧後に実行されるため、管理対象サーバの復旧時間で比較を行う。管理対象サーバの復旧処理として、ネットワークトラブルでは、対象サーバの再起動状態を確認後に強制起動を行い、ネットワーク機能を復旧する。サービストラブルでは、サービス提供プログラムの再起動で復旧せず、対象サーバを再起動することでサービス提供機能を復旧する。また、サーバ故障時において、クライアントからサーバに対する接続アクセスの影響を調査するため、クライアントから LB 経由で Mail, Web, Login のサービスを提供しているサーバに対して 1 秒間隔でアクセスを行う。

1 台 (M2) 故障時 (表 4 の上段部分) のサービストラブルにおいて、サーバ機能の復旧時間はサービスの停止を判断する時間 (T_s)、バックアップサーバのサスペンド状態からの起動時間 (T_{va}) と仮想 IP アドレスの設定時間 (T_{vv}) が主な要素となり、約 6.5 秒となっている。管理対象サーバの復旧時間ではサービスの停止を判断する時間 (T_s)、仮想サーバの再起動時間 (T_{vr}) と仮想 IP アドレスの設定時間 (T_{vv}) が主な要素となり、約 20.6 秒となっている。また、1 台 (M2) 故障時 (表 4 の上段部分) のネットワークトラブルでは、ネットワークの切断を判断する時間 (T_n)、バックアップサーバのサスペンド状態からの起動時間 (T_{va}) と仮想 IP アドレスの設定時間 (T_{vv}) が主な要素となり、約 5.4 秒となっている。管理対象サーバの復旧時間ではネットワークの切断を判断する時間 (T_n)、仮想サーバの再起動確認時間 (T_b)、ディスクキャッシュが影響する仮想サーバの起動時間 (T_{vsc}) と仮想 IP アドレスの設定時間 (T_{vv}) が主な要素となり、約 39.8 秒となっている。

表 4 RDBSS におけるサーバ故障時の復旧時間

(a) Characteristics for RDBSS: Problem in one server

Management server: W1	Recovery time (s) (Coefficient of variation)				Access disconnection time in client (s) (Coefficient of variation)
	Server function	Target server: M2			
Problem list	Compound operation	Compound operation	Single operation	Difference	
Service	6.549 (0.004)	20.838 (0.003)	20.626 (0.000)	0.212	0.000 (0.000)
Network	5.394 (0.008)	39.794 (0.001)	39.752 (0.006)	0.042	0.000 (0.000)

(b) Characteristics for RDBSS: Problem in two servers

Compound operation Management server: W1		Recovery time for target server (s) (Coefficient of variation)	
		Target server: L2	
		Service program	Network
Target server: M2	Service	22.315 (0.002)	41.332 (0.002)
	Network	20.873 (0.002)	20.947 (0.004)
		21.505 (0.005)	41.461 (0.002)
		39.862 (0.002)	39.836 (0.002)

1 台故障時における変動係数の結果から、各復旧処理は安定しており、それぞれの Difference の値には差が殆ど無いことが分かる。また、Mail サーバは冗長化構成のため、サーバ故障時においてもクライアントからサーバへの接続アクセスには影響していない。

2 台 (M2, L2) 同時故障 (表 4 の下段部分) では、M2 の復旧時間は 1 台故障時とほぼ同等となり、L2 は RDBSS による管理ファイルへの競合対策のため、約 1.5 秒の遅れが生じている。

1 台故障及び 2 台同時故障において、サーバ故障時や復旧処理において、PSS の誤動作などは見られなかった。

SIF 方式と単純統合方式における、プログラムの追加行数の比較結果を述べる。提案する SIF 方式を採用した場合、メインプログラムへの追加行数は PSS で 7 行、RDBSS で 2 行に留まった。これに対し、PSS と RDBSS を単純に統合した場合の追加行数は、PSS で 147 行、RDBSS で 88 行に達する。

5.3. 考察

本研究では、単独動作が可能な省電力サーバシステム (PSS) と高可用性サーバシステム (RDBSS) を複合動作させた、省電力かつ高可用性サーバシステムの正常動作を実証した。

本システムは PSS と RDBSS を独立して連携させる SIF 方式を採用しており、その構成上の優位性について考察する。PSS 及び RDBSS のメインプログラムはそれぞれ 146 行、238 行である。両システムを単純に統合した場合、PSS で 147 行 (約 100%増)、RDBSS で 88 行 (約 37%増) のプログラム行数が追加され、管理プログラムの複雑化を招く。一方、SIF 方式では各管理プログラムの独立性が維持され、「省電力機能」、「高可用性機能」、「相互制御機能 (SIF)」が分離される。これにより、バグ修正や機能拡張時の影響範囲を局所化でき、システム運用の効率化及びサーバ管理者の負担軽減に寄与すると言える。

一方、実用化に向けた課題として、単一障害点の排除が挙げられる。本実験システムではファイルサーバ (NFS) が単一障害点となる懸念があるため、実環境への適用には、高可用性ファイルサーバシステム[20]や Keepalived[21]を用いた冗長構成の採用が不可欠である。同様に、LB についても冗長化が必要となる。スケーラビリティに関しては、RDBSS において複数台の同時故障時に管理ファイルへのアクセス競合が発生するため、現在は排他制御を採用している。このため、本システムが想定する適用規模は数十台程度までとなる。より大規模な構成へ拡張する場合は、サーバ群を複数の管理グループに分割する階層的な管理手法の導入が有効である。

今後の展開として、より軽量かつ高速なコンテナ環境への移行を検討している。数百から数千のコンテナを管理するには Kubernetes[22]等の既存ツールが適しているが、本提案システムは数十個程度のコンテナ環境において、排他制御を伴う確実な制御を実現する手段として有効である。LB の設定や制御方法をコンテナ環境へ適応させることで、本システムの優位性を維持しつつ、負荷状況に応じた動的かつ柔軟なスケジューリング機能の実装が可能になると考える。

6. おわりに

単独で動作可能な専用の管理サーバを必要としない省電力サーバシステム (PSS) と高可用性サーバシステム (RDBSS) を複合動作させることで、専用の管理サーバを必要としない省電力かつ高可用性サーバシステムを開発した。PSS と RDBSS を複合動作させる場合の問題点を分析し、その詳細及び解決方法を示した。また、システムインタフェース方式を専用の管理サーバを必要としないサーバ管理方式に適用し、PSS と RDBSS における管理プログラムの動作効率を悪化せずに複合動作が可能となることを示した。本提案システムを導入した実験システムを構築し、PSS や RDBSS の単独及び複合動作時において、サーバシステムの構成変更やサーバ故障を再現する実験を行った。実験結果から、単独及び複合動作時の処理時間に差がなく、複合動作による省電力かつ高可用性サーバシステムが安定稼働することを示した。しかし、本研究での評価は実サーバ 3 台 (仮想サーバ 6 台) の小規模構成に留まっている。今後は、中小規模のオンプレミス環境での実用を想定し、数十個程度のコンテナシステムを構築して、負荷追従型のスケジューリング機能の追加及び動的な負荷変動に応じた省電力効率や故障頻度の変化が可用性に与える影響の定量的な調査などを検討課題としたい。

謝辞

本論文の執筆にあたり、査読者の方々に建設的かつ有益なご指摘やアドバイスをいただきました。ここに記して、心より感謝申し上げます。

参考文献

- [1] 総務省, “令和5年版情報通信白書第4章8節データセンター市場及びクラウドサービス市場の動向,” <https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/r05/pdf/n4800000.pdf>, 2024.5.15 参照.
- [2] 環境省, “地域脱炭素ロードマップ,” https://www.cas.go.jp/jp/seisaku/datsutanso/pdf/20210609_chiiki_roadmap.pdf, 2022.09.23 参照.
- [3] 環境省, “グリーン IT について,” https://www.env.go.jp/council/content/i_05/900425720.pdf, 2010.03.09 参照.
- [4] Kitamura, M., Takeshita, T., and Yamaguchi, T., “Development of Power-saving Server System not requiring a Specific Management Server,” JISSJ, Vol.16, No.2, 2021, pp.18-35.
- [5] Kitamura, M., “Configuring a Low-Cost, Power-Saving Multiple Server Backup System: Experimental Results,” IEICE Trans. Commun., Vol.E95-B, No.1, 2012, pp.189-197.
- [6] Kitamura, M., Takeshita, T., and Okazaki, T., “Development of Ring-type Dynamic Backup Server System not requiring Dedicated Management Server,” JISSJ, Vol.18, No.1, 2022, pp.22-42.
- [7] Kitamura, M., Shimizu, Y., and Tani, K., “Development and Operation Experiment of a Power-saving, High-availability Server System by Compound Operation of a Power-saving Server System and a Multiple-server Backup System,” JISSJ, Vol.15, No.2, 2020, pp.34-54.
- [8] 賣野豊, “フルメッシュ波長ルーチングで接続された FPGA を搭載した省エネルギー高性能サーバシステム,” 信学誌, Vol.105, No.11, 2022, pp.1313-1319.
- [9] 山浦隆博, 秋山晴彦, 小林優太, 橋康達, 山口健作, 田中信吾, 後藤真孝, 村井信哉, “ストレージからネットワークへ直接送信を行う低消費電力映像配信サーバ,” 信学論 B, Vol.J99-B, No.7, 2016, pp.501-514.
- [10] Xu, J., Wu, X., Huang, Q., and Sun, P., “How should the server sleep? – Age-Energy tradeoff in Sleep-Wake server systems,” IEEE Trans., Vol.7, No.4, 2023, pp.1620-1634.
- [11] Rout, S., Jaiswal, A., Bagade, M.K., and Patra, S.S., “SDN-based Mobile Edge Computing with Vacation of Virtual Machines,” IEEE Conference Proceedings, Vol.2024, No.INOCON, 2024, pp.1-6.
- [12] 奥村恭弘, 西村徹, 横関大子郎, 由良俊介, “第3回高可用性技術-24時間365日休まず運転するために,” NTT 技術ジャーナル, <https://journal.ntt.co.jp/backnumber2/0602/files/jn200602079.pdf>, 2023.6.14 参照.
- [13] Amazon EC2 Auto Scaling ユーザーズガイド, “Amazon EC2 Auto Scaling とは,” https://docs.aws.amazon.com/ja_jp/autoscaling/ec2/userguide/what-is-amazon-ec2-auto-scaling.html, 2022.02.04 参照.
- [14] Kubernetes, “水平 Pod 自動スケーリング,” <https://kubernetes.io/ja/docs/tasks/run-application/horizontal-pod-autoscale/>, 2022.05.09 参照.
- [15] Bukhsh, M., Abdullah, S., and Bajwa, I.S., “A decentralized edge computing Latency-Aware task management method with high availability for IoT applications,” IEEE Access, Vol.9, 2021, pp.138994-139008.
- [16] Saxena, D., Gupta, I., Singh, A.K., and Lee, C.-N., “A Fault Tolerant Elastic Resource Management Framework Toward High Availability of Cloud Services,” IEEE Trans. Network and Service Manage., Vol.19, No.3, 2022, pp.3048-3061.
- [17] 小野智央, 上田清志, “高可用分散クラスタにおける冗長性動的制御手法,” 信学論 B, Vol.J102-B, No.9, 2019, pp.705-709.
- [18] 田中龍馬, 北村光芳, 新城幸也, “複合動作による省電力かつ高可用性サーバシステムの開発,” 情報システム学会 第20回全国大会・研究発表大会, P009, 2024.11.
- [19] Kitamura, M., Tani, K., Takeshita, T., and Yamaguchi, T., “Development of Server Function Recovery System for Peer-to-Peer Method Server Management System adopted for Virtual Server System,” JISSJ, Vol.17, No.1, 2021, pp.27-34.
- [20] 北村光芳, 田中龍馬, “サーバ管理システムにおける高可用性ファイルサーバシステムの開発,” 情報システム学会学会誌, Vol.21, No.1, 2025, pp.28-40.
- [21] Keepalived.org, “Keepalived for Linux,” <https://www.keepalived.org/>, 2020.06.03 参照.
- [22] Kubernetes Documentation, “Kubernetes のコンポーネント,” <https://kubernetes.io/ja/docs/concepts/overview/>, 2022.05.09 参照.

著者略歴

北村 光芳（きたむら みつよし）

1984 年東京工芸大学工学部電子工学科卒。同年シャープ（株）入社。1990 年東京工芸大学助手，同大学講師を経て 2022 年より同大学准教授，現在に至る。博士（工学）。省電力，高可用性サーバシステムの構築，運用および計算機シミュレーションに関する研究に従事。

田中 龍馬（たなか りょうま）

2025 年東京工芸大学大学院工学研究科博士前期課程卒。同年（株）日立情報通信エンジニアリング入社。現在に至る。ストレージ製品の開発に従事。