高可用性システム統合監視機構の提案

敷田幹文　井口寧　藤枝和宏　松澤照男
北陸先端科学技術大学院大学 情報科学センター

近年，組織内ネットワークは大規模化・集中化し，電子化される業務範囲が拡大している．また，各業務の情報ネットワーク依存度が高くなり，サービスの中断を極力減らすための各種のハードウェアおよびソフトウェアを備えた高可用性サーバを運用している．しかし，機構が複雑な様々なサーバを運用するためには，障害の発生を検出し，管理者に通知する機構が極めて重要となるが，従来の監視システムは主としてネットワーク機器を対象としており，特に複雑な機構をもつ高可用性サーバには適していなかった．本論文では，大規模な組織において複数の高可用性サーバを統合的に監視するための新たな枠組みを提案し，これに基づいて設計したシステムについて述べ，本方式の有用性について議論を行う．

The Integrated Watch Mechanism for High Availability Servers

Mikifumi SHIKIDA　Yasushi INOGUCHI　Kazuhiro FUJIEDA　Teruo MATSUZAWA
Japan Advanced Institute of Science and Technology

Recently, network becomes large-scale in the organization. The range of business on the network expands. Moreover, the network dependency of each business rises. They introduce high availability servers which had the various hardware and software to reduce the interruption of the service. It is most important to detect occurrence of the fails. But, usual watch systems are used for watch of network equipments, and it was not suitable for the high availability servers which have especially complex mechanisms. In this paper, we propose the new scheme to watch high availability servers in the large-scale organization. And we state about the designed system based on the scheme, and we discuss about the usability of this method.

1　はじめに

近年，組織内ネットワークは大規模化・集中化しており，電子化される業務範囲が拡大し，データの量や重要度が増大している．その結果，各業務の情報ネットワーク依存度が高く，これまで以上に信頼性の高いシステムが求められている．そのため，大規模な組織では，サービスの中断を極力減らすための各種のハードウェアおよびソフトウェアを備えた高可用性サーバを運用している．

しかし，機構が複雑な様々なサーバを運用するためには，障害の発生を検出し，管理者に通知する機構が極めて重要となる．従来，ネットワーク機器などではSNMP (Simple Network Management Protocol) を用いた集中型監視システムが多く，異なるメーカ製品も監視可能な共通基盤として確立されつつある．また，トラフィックの軽減などのために単純な集中型ではない監視機構の研究もある [6, 11, 13, 12] が，主としてネットワーク機器を対象としている．これに対して，サーバ上のサービスに関してはそのような標準的機構が確立されていない．

特に高可用性サーバでは，ディスクドライブや各種インタフェースの故障と代替動作や，データバックアップシステムの挙動など，多種多様なイベントを扱う必要があり，SNMPを用いた機構など汎用の管理基盤もほとんどない．そのため，各サーバでは個々の製品毎の管理ツールを利用しており，正常稼動時にも1日に数十通のメールを受信するなど，管理コストの増大が問題となっている．

本論文では，大規模な組織において複数の高可用性サーバを統合的に監視するための新たな枠組みを提案し，これに基づいて設計したシステムについて述べるとともに，本方式の有用性について議論を行う．

以下，2章では，高可用性システムとそれを監視する従来の機構について述べて問題点を明らかにし，3章で，我々が提案する統合監視機構の設計法を述べる．4章では，本論文の方法の有用性についての議論を行う．

2 高可用性システムと監視機構

本章では，最近利用が広まってきた高可用性システムと，これまで用いられてきた監視機構について述べる．

2.1 高可用性システム

高可用性 (High Availability, 以下 HA と略す) システムとは，各種の障害やメンテナンスなどによって提供するサービスが中断する時間を，代替機や代替部品によって極力短くする機構を備えたシステムである．

通常は二重化 (もしくは多重化) によって実現されているが，本体全体の二重化だけでなく，各部品毎にも二重化されており，それらの各部を制御するために多数のソフトウェア部品から構成されている [10].

表 1: 著者の大学での集中管理対象ホスト概数

クライアントワークステーション	1,500
ワークグループサーバ	150
エンタープライズサーバ (高可用性システム 8 システムを含む)	30

我々の大学は大学院大学であり，ユーザの利用形態は端末室ではなく研究室内の自席での利用がほとんどである．そのため，我々の情報科学センターでは全学の各研究室内の個人用ワークステーションまで含めた大規模な集中管理を実施している [9]. 現在の集中管理対象ホストの概数を表 1 に示す．

このように大規模な集中化を行っているため，電子メールやファイル共有などの基本サービスを提供する各サーバの重要度はきわめて高く，ほとんどを高可用性システムとしている．しかし，表 1 に示したようにサーバの台数が多く，正常稼動時でも 1 日に数十通のメールを受信する状況で，障害の検出法が問題となっていた．

現在商品化されている高可用性システムは，各種のハードウェアおよびソフトウェア部品の集合体として構成されているものがほとんどである．そのため，例えば，ディスク装置内，ファイルシステム，ネットワークインタフェースなども管理方法，障害時の検出・通知法が異なっており，それぞれには高度の障害検出通知機構も備えていない部品が多い．

しかし最近では，各メーカは各部品とは独立した障害検出通知ソフトウェアを組み合わせて利用することが増えている．そのようなソフトウェアとして，SunCluster[2] の Management Center[5], MC/ServiceGuard[1] の EMS HA Monitors, JP1 Network Node Manager[7, 8] や SystemWalker[4] などがある．

2.2 サーバ監視機構の問題点

従来の監視ソフトウェアによるサーバ障害検出では，次のような問題がある．

1. サーバ外の状況がわからない
   検出ソフトウェアはサーバ上のソフトウェアメーカが提供するもので，サーバ上もしくはサーバの管理端末上にインストールされ，サーバ内の障害のみを対象としている．しかし，サービスの可用性を高めるためには，サーバだけでなく接続されるネットワークを含めて，クライアントから見たサーバへの到達性を調べなければいけない．
2. システム間の統一がとれない
   検出ソフトウェアは，各メーカで独自の商品であり，それぞれの間に互換性はなく，同一ソフトウェアであっても異なるサーバの情報を統合する機構はほとんどない．様々なサーバを少人数で管理する場合には，管理コストが増大する．また，組織内で開発されたシステムなど，各監視ソフトウェアが扱わないシステムからの情報を統合する必要がある．
3. 通知手段が不充分である
   障害発生時の通知方法としては，コンソールに

表示されるものが多く，メールを発信できても詳しい情報が得られない製品が多い．また，管理者は複数人で構成されるのが一般的で，サーバの種類や障害の程度によって通知方法が適切に選択されないと，障害情報の埋没化が起きる．

3 統合型監視システムの実現

我々は，組織内の各システムを監視し，その情報を統合して，障害発生時に管理者に通知するシステム TotalGuardian (以下 TG と略す) を開発した．

3.1 基本設計

以下のような基本方針のもとに TG を設計した．

1. モジュール化
   障害などのイベントの入力手段，および管理者への通知などの出力手段はそれぞれモジュール構成とし，容易に追加可能とする．
2. 他監視システムとの連携
   商用またはフリーソフトウェアの監視システムと相互にイベントを伝えることが可能とする．
3. 監視ホスト内ソフトウェアとの連動
   サーバ内で動作する各ソフトウェアと連動する．ポーリングを主とするネットワーク監視システムの中にも，各ソフトウェアへの対応がなされているものがある [7, 4] が，ログからの検出など受動的な連携がほとんどとなっている．
4. 複雑な情報の扱い
   単純な数値や文字列だけでなく，コマンド出力結果など複数行にわたるテキストを扱う．

3.2 基本構成

図 1 に，TG の基本構成の各モジュールを示す．多くの部分はスクリプト言語 Perl で記述した．TG の基本構成は単純なクライアント・サーバモデルであり，両者をそれぞれ「監視ノード」,「統合サーバ」と呼ぶ．両者の間は TCP ソケットを用いた独自プロトコルで通信を行う．この基本構成を複数組み合わせることによって，様々な形態に柔軟に適用が可能である．

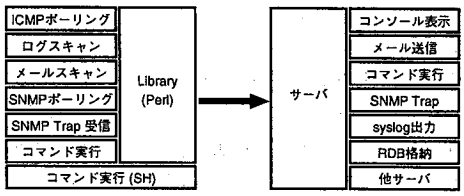

図 1: モジュール構成

3.3 監視ノード

図 1 の左側の監視ノードは，監視を行うホストである．監視対象ホストは必ずしも自ホストではなく，他ホストをポーリングしたり，他ホストからの Trap を受け付ける場合もある．組織内の各所に配置し，監視した情報を統合サーバへ送信する．

統合サーバへの通信プロトコルは単純なテキストの送信とし，重要度，ID，発生時刻，ホスト名，メッセージ，コンテキストなどを送る．単純なプロトコルとしたことにより，監視ノード側の動作環境の制限が緩まり，例えば Perl がなくとも標準的 UNIX コマンドの組み合わせだけでも記述が可能である．コンテキストとしては，イベント発生の要因を補足するための任意のテキストファイルを送ることができる．例えば，CPU 負荷が高い場合には ps コマンドの結果を付加したり，バックアップシステムのエラーの場合にはシステム全体の様々な状態を出力するレポートの結果を付加するような用途に利用できる．

監視方法には，ポーリングする場合とイベントの発生によって起動される場合の 2 つがある．監視対象としては，UNIX の syslog，一般のファイル，およびコマンドの実行結果などがある．SNMP Trap の受信には Net-SNMP パッケージを利用している．これらの入力の内容を解析して正規化するモジュールを作成する構造になっているため，様々な形態に対応可能である．また，サーバを構成する部品ソフトウェアの中には，特定の状況でのコマンド実行を組み込めるものが少なくないため，イベントの発生時に直ちに情報を取得できる．このことにより，サーバが持っている障害通知機構を本システムに連結して情報を収集することができる．

また，組織内各所のクライアントに監視ノードを置くことにより，ICMP や各サービスのプロトコルによるアクセスを試み，組織内各所のクライアントから見たサーバの稼動を調べることができる．

3.4 統合サーバ

図1の右側は，複数の監視ノードからのイベントを受ける統合サーバである．統合サーバには，「条件」とその場合の「通知先」の組を予め記述しておく．受信したイベントは，記述されている条件に適合した場合にのみそれに対応する通知が実行される．これにより，イベントの種類，重要度，発生ホスト名，時間帯など様々な要因によって適切な通知先を選択することができる．

通知先としては，syslog 出力，メール送信，SNMP Trap 発生などのほか，任意のコマンドが実行できる．また，統合サーバが監視ノードとなり，他の統合サーバにイベントを転送することも可能である．これを利用することによって組織内の監視機構の階層化など，様々な形態に対応することが可能となる．

4 ディスカッション

本章では，前章で述べたシステムを利用する際の構成法を述べ，著者らの組織内における試験運用について述べる．

4.1 TG を利用した構成例

TG では，図1に示した基本構成を元に様々な形態の監視体制を構築することができる．以下に，その構成例を述べる．

1. ネットワーク負荷の軽減
   大規模な組織では監視対象ホストの数が多く，それらへのポーリングによるネットワークトラフィックの増大が問題となっている [12]．文献 [11] と同様に階層に分けることによって負荷の軽減が期待できる．
   すなわち，各監視対象ホストをネットワーク的距離でグルーピングし，各グループの近傍にポーリングのための監視ノードを配置する．各監視ノードではポーリングの結果を全て統合サーバへ送るのではなく，障害発生時のみ上位の統合サーバへ送信することによって，大幅に負荷が軽減される．
2. 管理者階層に合わせた階層化
   大規模な組織では，多数の管理者が複数部署に分かれている場合がある．しかし，それらは独立ではなく，例えば，小規模な障害は支社内の管理部門が対応し，重要障害は本社内の管理部門が対応する場合がある．
   TG では，各部局の統合サーバより上位に組織全体の管理部門用の統合サーバを配置し，重要障害のみ上位に通知することが可能である．同様に3階層以上とすることも可能である．
   また，1部署の管理者が組織全体を集中管理している場合においても，各管理者の経験には差がある．例えば，頻繁に発生する小規模な障害に関しては複数の初級管理者が分担を決めて対応し，上級管理者は重要もしくは特殊な障害のみを担当することが考えられる．そのような場合には，統合サーバを分けて階層的に配置することにより，各自の担当する障害のみを収集するサーバが実現できる．
3. 監視機構の多重化
   各監視ノードや統合サーバ自身が障害のため動作を停止することもあり得る．商用監視システムの JP1 や SystemWalker などでは，全体のマネージャを高可用性サーバ上に置くことによってマネージャホストの障害時にも監視機構の継続を保証することが可能である．
   TG の場合には，各ホストを複数配置して多重化することによって可能となる．ただし，現在のところ，複数の監視ノードから届いた同一のイベントを1件とみなす機構は準備していないため，障害発生時には同一内容通知が複数件届くことになる．
4. 広域からの監視
   高可用性サーバが提供するような重要なサービスの場合には，1台の監視対象ホストに対して，組織内の各所の監視ノードから監視を行う．
   大規模な組織では，組織内であっても遠隔地のクライアントから多数のネットワーク機器を経由してサーバにアクセスしている．
   それらの個々のネットワーク機器の故障はネットワーク監視システムを用いて監視している場合であっても，全体を把握していない各末端管理者は故障機器を通過するサービスの重要度を把握しているとは限らず，その機器の重要性が分かりにくい．クライアント側から実際にサービスしているプロトコルで監視することによって，ユーザへのサービスに影響が出ているかど

うかが明確にわかる．

5. 商用システムとの連携
商用の監視システムを利用した場合，温度計やその組織独自の装置，独自開発アプリケーションなど商用監視システムが対応していないものは多い．商用システムでも多くのものは開発環境の提供 [3] や一部 API の公開を行っているが，それらの利用は一般に開発コストが高い．TG の機構はシンプルで柔軟であるため上記開発環境より低コストで対応可能である．また，TG から商用監視システムへ SNMP Trap や syslog 経由でイベントを伝えることも可能であるため，商用システムの簡易開発環境としても利用可能である．

6. 個別の監視システムの統合
サーバ上で稼動する各部品ソフトウェアの多くは，syslog や電子メールなどを用いて独自に障害の通知を行う機能を持っているものが多い．部分的に統合型監視システムとの連携が可能なサーバもあるが，各サーバの自社の監視システムに限定されていることが多く，複数メーカのサーバを導入している場合には各々に専用の監視システムが乱立するため，監視機構の統合の妨げとなる．このような場合には，TG を用いることによってそれらの統合が可能となる．

7. イベントの要約
統合サーバに集積されたイベントの履歴を参照し，その結果によって別のイベントを発生されることが可能である．それを自分自身もしくは他の統合サーバへ送信する．
これによって以下のようなことが可能となる．

- 複数イベントを要約した 1 つのイベントに置き換える
- 定期的に発生するイベントがないことによる異常を検出する
- イベントの頻度などを分析して変化を検出する

4.2 TG の運用実験

現在，我々の大学では，TG を一部のサーバの監視に試験的に導入している．これにより，以下のような実験を行っている．

1. ICMP ポーリング
ポーリングを行う例として，ICMP echo を用いて多数のホストの稼動確認を行うモジュールを作成した．

2. Sun Management Center 3.0 からのアラーム受信
商用監視システムとの連携例として，Sun Management Center 3.0 (以下 SunMC と略す) が検出し，アラーム通知を TG へ送るためのモジュールを作成した．このようなモジュールは 20 行程度の Perl スクリプトで記述できる．
このスクリプトを SunMC 側の通知の設定で起動するコマンドとして登録することによって連携が可能となる．

3. Sun Management Center 3.0 への通知
上記とは逆に，TG 側で検出したイベントを SunMC へ送信する．Java を利用した SunMC の開発環境 [3] を利用する他に，syslog を経由する連携も可能である．
ただし，SunMC は SNMP を想定した単純な値の蓄積を行うため，TG 内の全ての情報が伝わるわけではない．SunMC は障害の発生状況をまとめたデータベースとして利用し，ここの障害に関する補足情報は TG 内を参照するという利用形態となる．

4. 障害通知メールの受信
我々が過去に作成したソフトウェアは障害をメールで通知するものが多い．また，近年のサーバ機は，OS が稼動する CPU とは独立した稼動監視ハードウェアを備えている機種もある．これはやはり独立したネットワークインタフェース経由で本体の異常を通知するもので，電子メールを利用している．
これらが送信する電子メールを処理し，障害の種類や重要度を特定するモジュールを作成した．このモジュールを備えた監視ノードをメールサーバ上に置いて，各種のメールの受信を行わせている．

5 おわりに

本論文では，大規模な組織において高可用性サーバを含む多数のサーバを統合的に監視するための新たな枠組みを提案した．また，これに基づいて実現したシステムについて述べ，本方式の有用性

に関する議論を行った．組織内のサーバは，今後ますます大規模化・集中化し，信頼性の高い高可用性サーバの導入によって複雑になるが，本論文で提案するような障害情報の統合管理によって管理コストの削減が期待できる．

これまでは試験的な運用のみであるが，今後は，既存の監視システムとの連携を行い，実運用している各サーバを全て本システムの管理下に置き，実際の障害発生時などの日々の運用業務を通して評価する予定である．

参考文献

[1] HEWLETT-PACKARD COMPANY. *Managing MC/ServiceGuard*, 1998.

[2] SUN Microsystems, Inc. *SUN Cluster2.2 System Administration Guide*, 1999.

[3] Sun Microsystems, Inc. *Sun Management Center 3.0 Developer Environment Reference Manual*, Nov. 2000.

[4] Systemwalker/centricmgr version 10. http://systemwalker.fujitsu.com/jp/cen/.

[5] サン・マイクロシステムズ (株). Sun Management Center 3.0 ソフトウェアユーザーマニュアル, Jan. 2001.

[6] 浜田雅樹, 藤崎智宏, 犬束敏信, 蔭山克禎. インターネットにおける協調管理プラットフォームの提案. 情報処理学会 分散システム/インターネット運用技術報告 DSM-10, pp. 31–36, Jul. 1998.

[7] (株) 日立製作所. 統合ネットワーク管理システム ネットワークノードマネージャネットワーク管理ガイド, Jun. 2000.

[8] (株) 日立製作所. 統合ネットワーク管理システム サーバシステム管理, 第 2 版, May. 2001.

[9] 敷田幹文, 井口寧, 丹康雄, 松澤照男. 大規模分散システムの集中運用管理における効率化技術の提案. 情報処理学会 分散システム/インターネット運用技術シンポジウム, pp. 75–80, Feb. 1999.

[10] 敷田幹文, 井口寧, 三輪信介, 丹康雄, 松澤照男. 大規模高可用性サーバの設計と運用. 情報処理学会 分散システム/インターネット運用技術シンポジウム, pp. 57–62, Feb. 2001.

[11] 長田智和, 谷口祐治, 玉城史朗. 大規模分散ネットワーク運用管理システムの提案. 情報処理学会 分散システム/インターネット運用技術報告 DSM-20, pp. 31–36, Dec. 2000.

[12] 三好優, 釜洞健太郎, 朴容震, 浦野義頼, 富永英義. モバイルエージェントによる大規模・分散形ネットワーク管理法の一提案. 情報処理学会 マルチメディア通信と分散処理報告 DPS-97, Mar. 2000.

[13] 知念眞也, 長田智和, 谷口祐治, 玉城史朗. 分散オブジェクト技術を用いたネットワーク監視システムの設計. 情報処理学会 分散システム/インターネット運用技術報告 DSM-20, pp. 37–42, Dec. 2000.