

インターネットサービスの特徴を考慮した サーバ稼動監視システムの実装

伊藤 武[†] 磯川 弘実[‡] 萱島 信[‡] 寺田 真敏[‡]

日立中部ソフトウェア(株)[†]

(株)日立製作所 システム開発研究所[‡]

要旨

企業における業務遂行のために、インターネット/イントラネットを基盤とするメール、WWWなどのインターネットサービスが、情報交換インフラとして重要な役割を担うようになった。これに伴い、インターネットサーバの管理者は、サーバに障害が発生した事を早期に発見すると共に、障害復旧を迅速に行う必要がある。そこで、我々は、インターネットサービスの稼動監視を行う監視システムを開発した。監視システムを開発する上で、サーバの状態を取得し障害が発生しているかを判断するプログラムを数多く開発する必要があるため、工数の増加が問題となってくる。本稿では、監視システムをオブジェクト指向設計技法で開発し、工数を削減した実装方式について述べ、その評価を行う。

Implementation of a server surveillance system that depends on the feature of the Internet services

Takeshi ITO[†] Makoto KAYASHIMA[†] Masato TERADA[†] Hiromi ISOKAWA[‡]

Hitachi Chubu Software, Ltd.[†]

Systems Development Laboratory, Hitachi, Ltd.[‡]

In an enterprise, the Internet/Intranet came to be an important information infrastructure for performing works. Therefore, the administrator of the Internet servers needs to discover obstacles occurred in the servers in its early stages, and needs to perform obstacle restoration quickly. For this reason, We propose a surveillance system for discovering the server obstacles. We have a problem that has high cost for development. In this paper, We propose the system that use Object Oriented design, report that implemet the system for low cost, Finally we evaluate this tool.

1.はじめに

現在、多くのユーザが企業/個人の間で情報の提

供、交換、取得を目的に、メール、Web等のインターネットサービスを利用している。

インターネット/イントラネットを使用した様々な

サービスが提供されるようになってきたため、インターネットサービスが稼働しているサーバであるインターネットサーバの台数は増加し、それに伴いサーバ管理者の負担も大きくなっている。そこで我々は、サーバ管理者が行う管理作業の負担軽減を目的にインターネットサーバ稼働監視ツールを開発している。

監視対象であるインターネットサービスはサービス実行環境の多様化しているため、サーバ稼働監視システムの開発工数が増加するという問題が発生する。本稿では、監視対象であるインターネットサービスの稼働監視を行うプログラムを開発する上で、開発工数の増加という問題点を解決する監視システムの設計方式と実装した監視システムについて述べる。

2. インターネットサーバ稼働監視システム

2.1 インターネットサービスの特徴

メールや WWW 等のインターネットサービスは、現在様々な OS 上で利用されており、多くのベンダーやフリーのアプリケーションが提供されている。また、それらのアプリケーションは、機能向上やセキュリティ向上のため頻繁にバージョンアップが行われ、サービスプログラムの設定はサイト管理者が自由にカスタマイズできることが多い。

このため、インターネットサービスはその実行環境が多様化しており、まったく同じ環境のサーバはまず存在しないという特徴を持つ。

また、インターネットサービスが正常に稼働しているかどうかは、従来のようにサーバの生き死にのみでは判断できない。例えば、ニュースサービスでは、サーバが稼働し、プロセスが動作していても、ニュースプールディレクトリが一杯でニュースサービスが正常に稼働できていないことがある。このため、インターネットサービスは、稼働状況を把握するためにサービスを提供して

いるプロセスが稼働していることを確認するだけでなく、サービスが使用しているリソースが足りていて、かつ、他サービスによる障害の影響が起こっていないことを確認する等の作業を必要とする特徴もある。

そこで、我々はこのようなインターネットサービスの特徴を考慮したインターネットサーバ稼働監視システムを開発している[1]。本システムは、(a)マネージャ(b)エージェントで構成されている。

(a)マネージャ

- ・管理ホスト上で動作するプログラムで、制御モジュールとデータ加工モジュールにより構成される。
- ・制御モジュールは後述するエージェントと通信し、監視データを集める機能を持つ。
- ・データ加工モジュールは、収集した監視データを管理者が定めた警告値によって報告する機能と、時間毎による監視対象サーバの状態変化の推移をグラフ化する機能を持つ。

(b)エージェント

- ・エージェントは監視対象サーバ上で動作するプログラムで、監視モジュールと制御モジュールにより構成される。
- ・監視モジュールはインターネットサービスの稼働状態を調査する機能を持つプログラムで、リソース性能、プロセスの生き死になど多様な監視項目について1つずつ作成する。
- ・監視モジュールは、OS とのインタフェースを担当する状態取得部と取得したデータの解析を担当する障害判断部により構成される。
- ・制御モジュールは監視モジュールの起動およびマネージャとの通信をする機能を持つプログラムである。

インターネットサーバ稼働監視システムの構成を図1に示す。

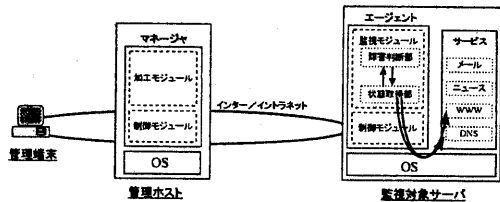


図1 インターネットサーバ稼働監視システムの構成

2.2 インターネットサーバ稼働監視システム開発における課題

2.1 節で述べたように、インターネットサービスはサービス実行環境が多様化しているため、監視対象サーバ上で動作するエージェントを作成する上で以下の2つの問題点が発生する。

- ・監視対象であるサービスアプリケーションの種類は豊富であり、各々のアプリケーションは機能の追加やセキュリティの向上によりバージョンアップが頻繁に行われる。このため、様々なサービスの監視を行うには、色々な監視モジュールを多数作成する必要がある。
- ・インターネットサービスの稼働状況の把握は、サーバのリソースやプロセスの生き死にの状況を、OS に対して問い合わせることにより実現する。しかし、監視対象のサーバは様々な OS で動作しており、それぞれの OS により上記の情報の取得方法が異なるため、監視モジュールは OS 毎に異なった動作を実行できる必要がある。

以上の2つの問題より、インターネットサーバ稼働監視システムの開発には、エージェント内にある監視モジュールの開発工数を削減することが特に重要な課題となる。

3.監視プログラムの実現方式の提案

3.1 基本設計

監視モジュールの開発工数がかかるのは、同様の機能を持つ監視モジュールを様々な OS に対して作る必要があることが原因となっている。

そこで、オブジェクト指向設計技法[2]を監視モジュールの設計に取り入れ、オブジェクト指向の特徴である継承を用いて (1) 各監視モジュールで共通する処理を基底クラスのメソッドとして定義し、(2) 個々に異なる処理を派生クラスのメソッドとする ことで開発工数を削減することにした。

具体的には、監視モジュールの状態取得部は、(1)障害判断部に障害判定を依頼する処理 と (2)OS よりデータを取得する処理 を持つ。そこで、状態取得部を1つの状態取得オブジェクトクラスとし、OS に依存しない前者の処理は基底クラスのメソッドとして定義し、OS に依存する後者の処理は派生クラスのメソッドとして定義した。

また、監視モジュールの障害判断部は、(1)状態取得部からサーバの状態を取得する処理 と (2)取得したサーバの状態から監視項目に障害が起きているかを判断する処理 を持つ。そこで、障害判断部を1つの障害判断オブジェクトクラスとし、サービスに依存しない前者の処理を基底クラスのメソッドとして定義し、後者の処理を派生クラスのメソッドとして定義した。

このクラス構成を UML(Unified Modeling Language)記法[3]で表現すると図2、図3になる。影付きのクラスは、監視モジュールが生成し、使用するオブジェクトのクラスである。

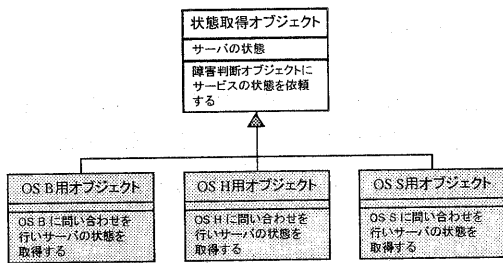


図2 状態取得オブジェクトクラス

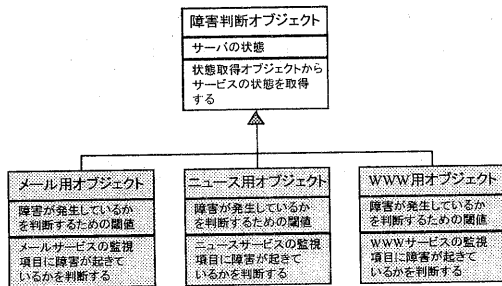


図3 障害判断オブジェクトクラス

監視モジュールは、以下のように動作する。

- (1) 監視するサービスに対応する障害判断オブジェクトクラスの派生クラスのインスタンスを生成する
- (2) 障害判断オブジェクトは、監視対象サーバのOSに応じた状態取得オブジェクトクラスの派生クラスのインスタンスを生成する
- (3) 状態取得オブジェクトはOSに問い合わせるメソッドを発行し、サーバの状態を取得する
- (4) 状態取得オブジェクトは障害判断オブジェクトにサーバ状態のデータを依頼する
- (5) 障害判断オブジェクトの取得したサーバ状態を加工するメソッドを発行する

例えば、OS B上で動作するメールサービスの監視を行う場合は図4のように動作する。影付きのオブジェクトは、実際に作成されるオブジェクトである。

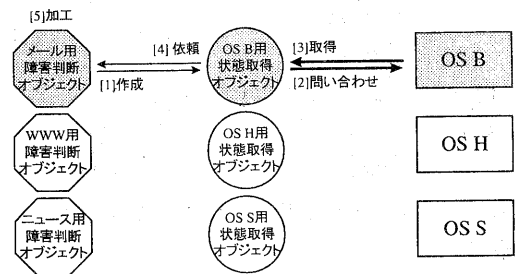


図4 監視モジュールの動作

3.3 拡張性の検討

本節では、本方式による開発工数の削減効果を検討する。

例として、OSのバージョンアップにより、OSに対する問い合わせの方法が変更された場合を考えてみる。

従来の非オブジェクト指向システムで新規バージョンに対応した監視モジュールを設計すると、新規バージョンに対応したOSへの問い合わせ処理を記述する他に、各バージョンに対応させるため、既存バージョンと新規バージョンを振り分ける条件分岐を追加する必要がある。これは、既存モジュールに手を加えなければ実装できないため、開発ステップ数が増加したり、保守性の確保が難しくなる。

しかし、監視モジュールをオブジェクト指向で開発すると、新規バージョン用のクラスは既存バージョン用のクラスの派生クラスとして作成される。派生クラスには新規バージョンでOSに対する問い合わせの方法が変更されたメソッドのみを定義する。新しいバージョンに対応した時のクラス構成をUML記法で表現すると図5になる。

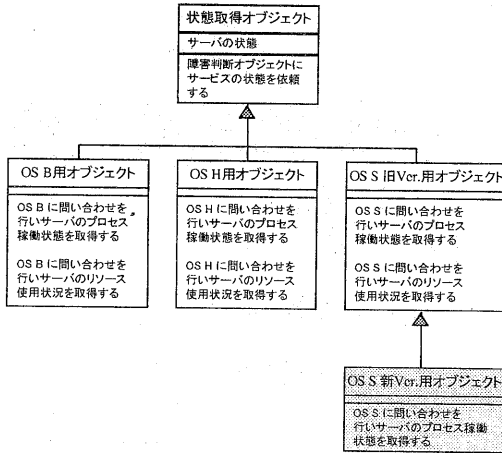


図5 新しいクラス構成

以上のように設計することで、OS に対する問い合わせの方法が変更されたメソッドを使用する場合は、新規バージョンに対応したメソッドが既存バージョンに対応したメソッドをオーバーライドし、新規バージョンに対応したOS に対する問い合わせを行う。また、既存バージョンとOS に対する問い合わせ方法が同じメソッドを使用する場合は、既存バージョン用のクラスのメソッドが継承され、OS に対する問い合わせを行う。このように、監視モジュールの拡張による新規開発は、既存バージョンのメソッドでは動作しない処理のみを開発するだけで良くなり、従来のプログラムについては変更を加える必要がない。

OS のバージョンアップによるオブジェクト指向言語で開発したときと非オブジェクト指向言語で開発したときの工数の差を(表)に示す。オブジェクト指向言語 perl5、非オブジェクト指向言語 perl4 で同機能の監視モジュールを1つ作成した。

表1 拡張時の開発工数

	perl4	perl5
OS 拡張時	1.54k ステップ	0.72k ステップ

以上の結果から、監視モジュールをオブジェクト指向で開発することにより、拡張による工数の増

加を最小限に抑えることが可能となったといえる。

4. プロトタイプシステム

我々は、前章で述べたインターネットサーバ稼働監視システムの設計方式を基に、プロトタイプシステムを開発した。

本プロトタイプでは、以下の機能を実現した。

- ・インターネットサービスの中で、メール、WWW、ニュース、DNS を対象とした。監視項目を表2に示す。
- ・インターネットサービスの稼働状態は、プロセスの有無、ネットワークを介した応答により監視し、異常があった場合にはサーバ管理者に報告する。
- ・インターネットサービスが使用するディスクリソースの空き容量とiノードテーブルの空き容量を監視し、閾値を超えて不足した場合には管理者に報告する。特に、メールサービスについては、ユーザ毎のメールドロップ使用量も監視する。
- ・インターネットサービスが出力するログを解析し、管理者が指定した文字列が検出された場合には管理者に報告する。
- ・メールキューやメールスプール、ニューススプールが使用するディスクリソース使用量の1日、1週間、1ヶ月間の変化をグラフにして、管理者に報告する。

表2 実装した監視項目

監視項目	詳細説明
サービスプロセスの状態	メール、WWW、ニュース、DNS の各サービスについて、プロセスが存在するかを検査

ディスクリソースの状態	メール, WWW, ニュース, DNS の各サービスについて, 使用するディスクリソースとiノードテーブルの空き状況を検査。各サービスで監視する項目は以下の通り。	
	メール	キュー、スプール、ログ、テンポラリ
	ニュース	スプール、ログ、テンポラリ
	WWW	キャッシュ、ログ
	DNS	バックアップ、ログ
サービス応答状態	ネットワークを介したアクセスにサービスプロセスが正しく応答するか検査	
ログ出力の状態	各サービスで出力されるログを解析し、異常報告がされているかを検査	
サービスの時間推移状態	ディスクリソース使用量、ネットワークパケットの入出力量、CPU 使用率について、単位時間あたりの値をグラフ化して報告	

本プロトタイプで開発した状態取得クラスのクラス階層と WWW サービス用の障害判断クラスのクラス階層を UML 記法で表現すると図 6、図 7 になる。

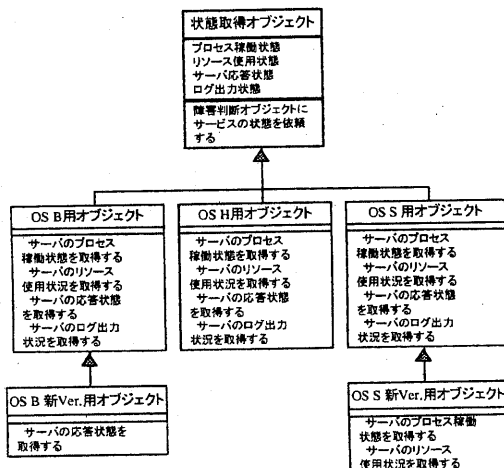


図6 実装した状態取得オブジェクトクラス

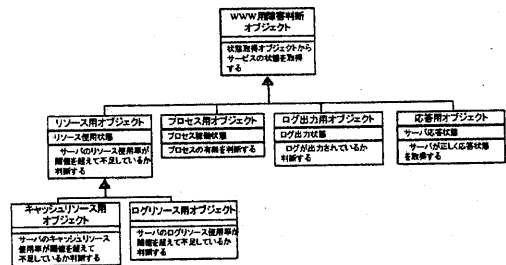


図7 WWW 障害判断オブジェクトクラス

5. まとめと今後の課題

インターネットサーバの稼働監視を行うプログラムは監視項目の多様化により開発工数を抑えることが特に重要となる。そこで、インターネットサービスの監視処理を行う監視モジュールをオブジェクト指向設計技法を用いることにより、

- (1) OS やサービスによって動作が共通である部分のメソッドを基底クラスに定義し、少しずつ異なる動作部分のメソッドをを派生クラスに定義することで、工数を低減
- (2) OS のバージョンアップによるメソッドの変更部分を、差分のみの開発により工数を最小限に抑えることが可能

となり、インターネットサーバ稼働監視システムの開発工数を抑えることができ、本方式の有効性を確かめた。

今後は、2.1 節で述べたサービスプログラムの設定はサイト管理者が自由にカスタマイズできることに対する課題が解決できていないため、この問題を解決できるサーバ稼働監視システムを作っていきたいと考えている。

参考文献

- [1] 磯川 弘実：インターネットサービスの特徴を考慮したサーバ稼働監視システムの提案,98-DSM-9,pp.1-6,1998
- [2] I.グラハム：オブジェクト指向概論第2版,トッパン
- [3] <http://www.rational.com/>