

Proxy 技術を利用した Web サービスのための プラットフォームの提案

坂本 暁 北 英彦 高瀬 治彦 林 照峯
三重大学工学部

Web の普及によって、Web ページの翻訳、有害 Web ページのブロックなどの Web サービスに対するニーズが高まっている。このような Web サービスを実現する方法として、Proxy を利用して実現する方法がある。この方法は Web ブラウザの種類に依存せずにサービスが利用できるなどの利点がある。しかし、Proxy 技術を利用した Web サービスの従来の実装方法では、一つのサービスにつき一つの Proxy を必要とするため、利用者にとっては複数のサービスの利用が困難、開発者にとっては新規サービスの開発コストが高い、という問題点がある。本研究では、これらの問題点を解決するために、一つの Proxy で複数の Web サービスを提供できるプラットフォームを提案する。

Platform for Web Services using Proxy Server

Satoru Sakamoto, Hidehiko Kita, Haruhiko Takase, and Terumine Hayashi
Faculty of Engineering, Mie University

In this paper, we propose a platform for proxy-server based Web services that include automatic language translation, blocking harmful Web pages and so on. The features of the platform are as follows:
(1) It is easy to use multiple services for users. (2) It is easy to develop a new service for programmers.

1. はじめに

Web の普及によって、Web ページの翻訳、有害 Web ページのブロック、ウィルスの除去などの Web サービスに対するニーズが高まっている。このような Web サービスを実現する方法として、Proxy を利用する方法がある。

Proxy とは Web ブラウザと Web サーバの間での HTTP 要求・応答を中継するシステムである。Proxy を設置する本来の目的は、インターネットからのアクセスに対してローカルネットワークのセキュリティを確保すること、HTTP 応答のキャッシングによってネットワークのトラフィックを軽減することである。

Proxy は HTTP 応答を中継しているので、中継途中にその HTTP 応答の操作が可能である。これを利用して、先にあげた Web ページの翻訳などの

Web サービスを実現することができる。Proxy を利用して実現できる Web サービスを、本論文では、Proxy 技術を利用した Web サービスと呼ぶ。

既の実現されている Proxy 技術を利用した Web サービスは、機能の固定したそのサービス専用の Proxy を開発する方法で実装されている。しかし、この方法では、複数のサービスを利用するためには、複数の Proxy を多段に利用する必要があり、複数のサービスの利用が難しい。また、サービスの開発者にとっては、単純なサービスであっても Proxy 全体を製作しなければならないため、サービスの開発コストが高いという問題がある。

Proxy を利用した Web サービスの一種として、Web ページへの情報付加サービスがある。服部ら [1]の研究では、情報付加サービスの開発を容易にするために、一つの Proxy で複数のサービスを

提供できるプラットフォームの提案を行っている。

本研究では、一つのProxyで複数のサービスを提供する服部らの考えを採用し、上記で述べたProxy技術を利用したWebサービスの従来の実装方法の問題点を解決するためのプラットフォームを提案する。

2. Proxy 技術を利用した Web サービス

Proxy 技術を利用した Web サービスは、Proxy が中継途中の HTTP 要求・応答を操作できることを利用している。また、Proxy の持つキャッシュとアクセスログ、同じProxyを利用している利用者の状況を利用するサービスも提案されている[2]。さらに、利用者からの HTTP 要求以外の入力、他のアプリケーションとの連携を利用したサービスも実現されている[1]。これらをまとめた Proxy 技術を利用した Web サービスのモデルを図 1 に示す。

また、Proxy が HTTP 要求・応答を操作できることはすでに述べたが、HTTP 要求・応答の流れも操作できる。図 2 に示す流れの操作を利用したサービスが実現されている。

- (a) 通常の Proxy の中継動作である。
- (b) Web サーバに要求を中継せずに Proxy が応答を返す動作ある。キャッシュを利用したネットワークのトラフィックの削減サービスなどがこの流れである。
- (c) 通常の中継動作の後に複数の Web サーバとの通信を行う流れである。Web ページの先読みサービスなどがこの流れである。

図 1, 図 2 に示すように、Proxy の持つアクセスログとキャッシュ、利用者の状況、そして利用者からの入力、他のアプリケーションの連携を利用して、HTTP 要求・応答、また、その流れを操作することで実現できるサービスを、本論文では Proxy 技術を利用した Web サービスと呼ぶ。

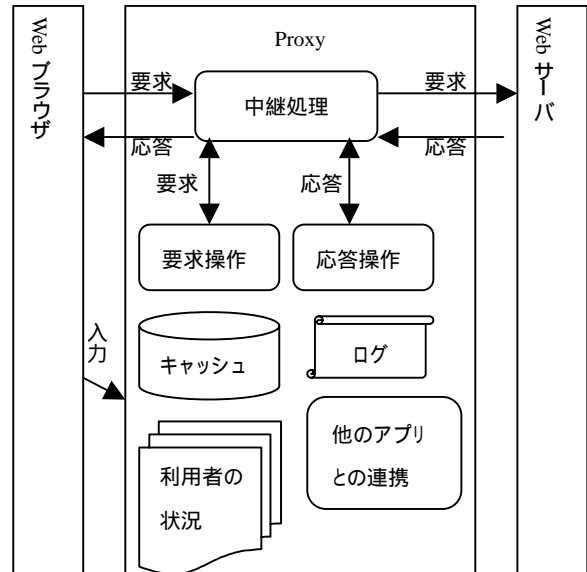


図 1: Proxy 技術を利用した Web サービスのモデル

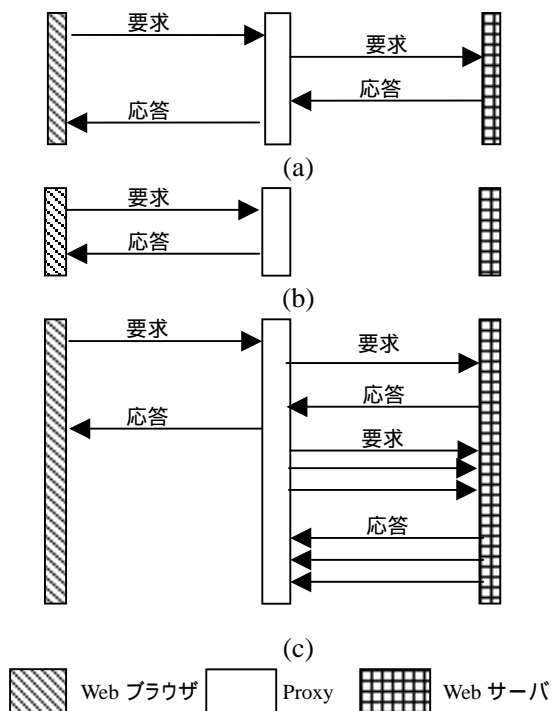


図 2: HTTP 要求・応答の流れの操作

3. Proxy 技術を利用した Web サービスの従来の実装方法とその問題

Proxy 技術を利用した Web サービスの従来の実装方法は、図 3 に示すように、Proxy 本来の機能のための中継処理とサービス本来の処理が一体化した構成をしている。基本的に一つのサービ

スにつき一つの Proxy の製作を必要とするため、(1) 利用者にとって複数のサービスの利用が難しい、(2) 開発者にとって新規サービスの開発コストが高い、という問題がある。

(1)については、サービスごとに専用の Proxy を用意して提供する従来の方法では、複数のサービスを利用することは複数の Proxy を多段に利用することになり、それぞれの Proxy に対する設定作業が必要であるため、複数のサービスの利用が難しくなる。

(2)については、Webサービスの開発において、サービス本来の処理の他に、HTTP 要求と応答の中継処理を作る必要があるため、単純なサービスであっても開発コストが高くなる。

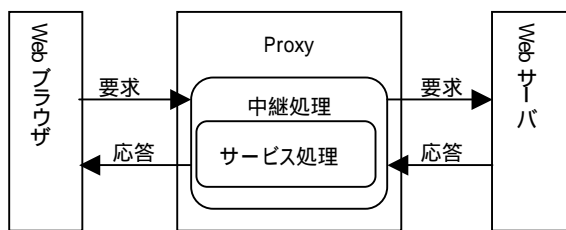


図 3:従来の実装方法

4. プラットフォームの提案

4.1. 要求仕様

本研究では、前章で述べた問題を解決するために、Proxy を利用した Web サービスのためのプラットフォームを提案する。提案するプラットフォームは前章で述べた問題を解決できるように、以下の要求を満たす必要がある。

- (1) 複数のサービスを容易に利用可能
- (2) 新規サービスを容易に追加可能
- (3) 新規サービスを容易に開発可能

これらを実現するための基本的なアプローチとして、要求(1)(2)に対してはプラットフォームのシステム構成を工夫することによって実現する。要求(3)に対しては、各サービスに共通する部分をなるべくプラットフォームによって提供することで実現する。

4.2. システム構成

要求(1)の実現方法について述べる。一つのサービスにつき一つの Proxy を用意する従来の実装方法では、複数のサービスを利用するためには複数の Proxy の利用が必要であったため、設定が複雑になり、複数のサービスの利用を困難にしていた。したがって、一つの Proxy で複数のサービスを提供することができれば、複雑な設定が不要になり、複数のサービスの利用が容易になる。

次に、要求(2)の実現について説明する。従来の実装方法では中継処理とサービス処理が一体化している。この中継処理とサービス処理の関係が密接なことが新規サービスの追加を困難にしている。したがって、要求(2)を実現するためには、中継処理とサービス処理を分離し、両者の間のインタフェースを決めればよい。これによって、新規サービスを追加することができるようになる。

さらに、一つの Proxy で複数のサービスを提供するので、利用するサービスは利用者ごとに選択できるようにしなければならない。そのためには、先ほど分離した中継処理とサービス処理の間に接続管理機能を挟み、サービス選択機能を用意すればよいと考える。サービス選択機能が利用者からの利用サービスの選択を受け付け、接続管理機能が利用者の選択したサービスを図 4 に示すように順番に適用するようにすれば、利用者ごとに利用するサービスを選択できるようになる。

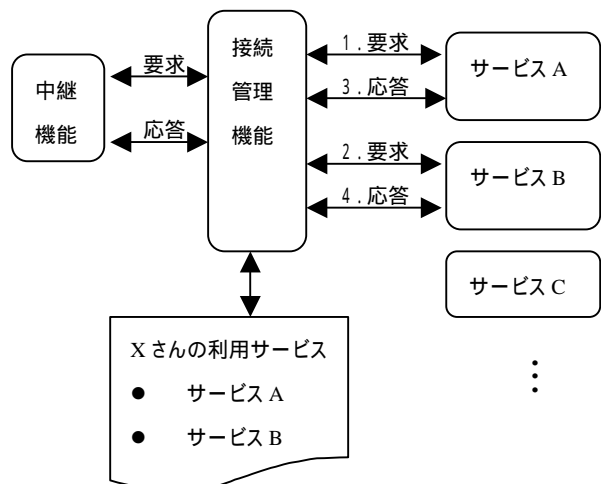


図 4:接続管理機能の動作

4.3. 共通部分の提供

要求(3)を実現するには、共通機能をなるべくプラットフォームによって提供し、サービス開発者が作成する部分を少なくすればよい。

プラットフォームによって中継処理などの機能は提供されているので、サービス開発者はサービス本来の処理だけを作ればよい。そこで、サービスの開発を容易にするために、サービスプログラムの雛型を提供する。

まず、図 4 に示す構成にしたことで、すべてのサービスプログラムは接続管理機能との通信機能を含む必要があるので、これを雛型で提供する。

次に、サービスプログラムで利用する可能性がある典型的な処理をライブラリとして提供する。図 1 と図 2 で示した Web サービスのモデルから明らかのように、Web サービスは、キャッシュ、アクセスログ、利用者からの入力、他のアプリケーションとの連携を利用する可能性がある。また、Web サービスにおいて、要求・応答及びその流れを操作する可能性がある。そこで、図 5 に示すように、キャッシュ・ログ・利用者の状況の利用機能、HTML 文書に関する典型的な処理など、をライブラリとして提供する。入力利用機能については Web サーバの機能を提供し、CGI などを用いて記述できるようにする。他のアプリケーションとの連携はサービスプログラムを実装するとき、そのアプリケーションを呼び出すことで可能である。

要求・応答の操作処理はサービスに固有なので、サービスの雛型ではインターフェースのみを提供する。要求操作処理のインターフェースは入出力ともに HTTP 要求、応答操作処理のインターフェースは HTTP 応答にする。インターフェースは決められているので、サービス開発者は要求・応答操作処理の処理手続きだけ記述すればよい。

要求・応答の操作処理を概念レベルで記述できるようにするために、HTTP 要求・応答をオブジェクトとして提供する。

以上で述べた実現手法をまとめるとシステム構成は図 6 のようになる。

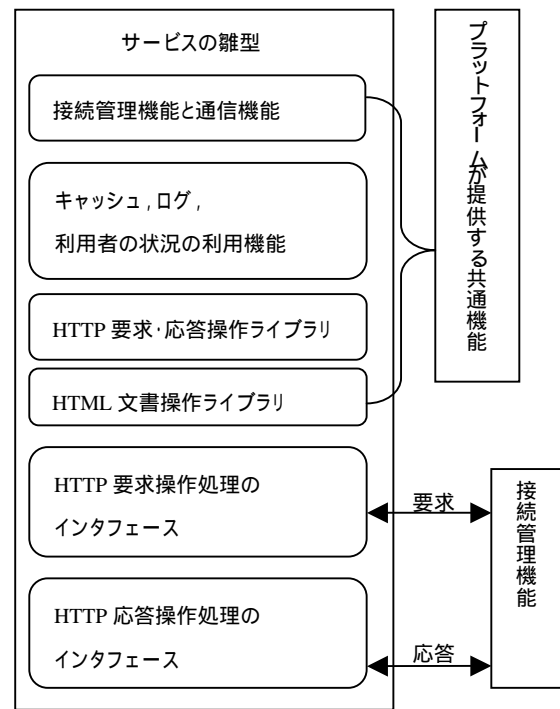


図 5: サービスの雛型

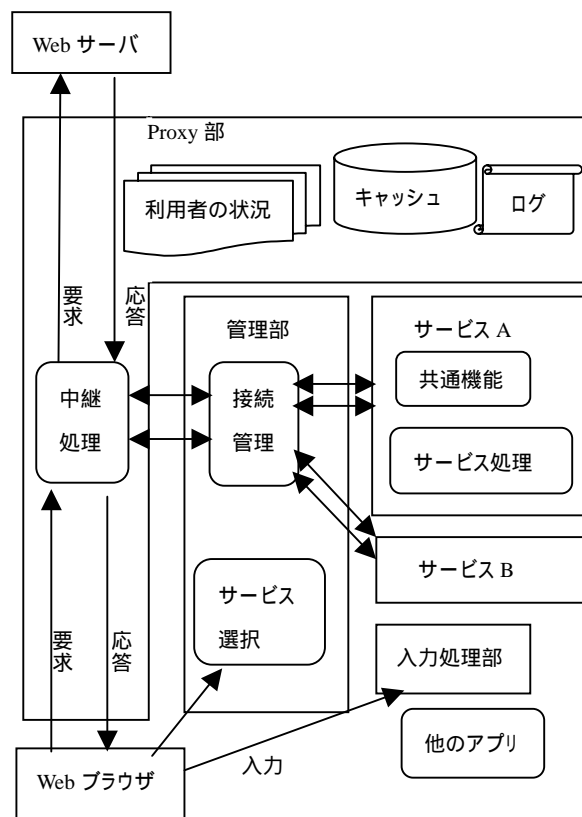


図 6: システム構成

5. プラットフォームの実装

提案する Proxy 技術を利用した Web サービスのためのプラットフォームの実装機能について述べる。

5.1. Proxy 部

Proxy 本来の機能である HTTP 要求と応答の中継などを担当する。以下の機能を持つ。

- HTTP 要求と応答の中継機能
- 中継途中の HTTP 要求と応答を管理部に渡す機能
- Proxy 認証を利用した利用者の認証機能
- アクセスログを記録する機能
- キャッシュ機能
- 先読み機能

5.2. 管理部

中継処理とサービスとの接続管理と利用者の管理などを担当する。以下の機能を持つ。

- 接続管理機能
- サービス選択機能
- 利用者の管理機能
- サービスの管理機能
- HTML 文書の解析

5.3. サービス

サービスは実際のサービス処理を担当する。システムを稼動させたままサービスの追加を行えるようにするために、サービスは Proxy 部、管理部とは独立したプロセスであり、また、各サービスも相互に独立したプロセスである。管理部とは TCP/IP を用いて通信する。開発者はサービスの雛型を利用してサービスを開発する。雛型はオブジェクト指向言語の Java のクラスとして実装した。サービスの雛型は以下の機能を持つ。

- 管理部との通信機能
- サービスの登録機能
- サービスプログラムへのプログラムインタフェースの提供
- HTTP 要求・応答操作のライブラリ
- サービス処理の監視機能
- キャッシュとログの利用機能
- Web サーバ機能

5.4. 入力処理部

入力処理部は利用者からサービスへの HTTP 要求以外の入力処理を担当する。利用者はサービスを利用するために Web ブラウザを利用しているので、入力操作は Web ページ上から行うのが自然と考えられる。そこで、既存の Web サーバを用意し、CGI もしくは JavaServlet を用いて入力処理を行い、ファイル、データベースなどを介してサービスに入力を引き渡せるようにする。

6. サービス作成例

提案したプラットフォームの有効性を確かめるために Web サービスの試作を行った。

6.1. キーワード強調表示サービス

Web ページ内の指定したキーワードを強調表示するサービスである(図 7)。指定したキーワードが強調表示されることで、大量の Web ページで情報を検索中に探している情報に気づきやすくなる。

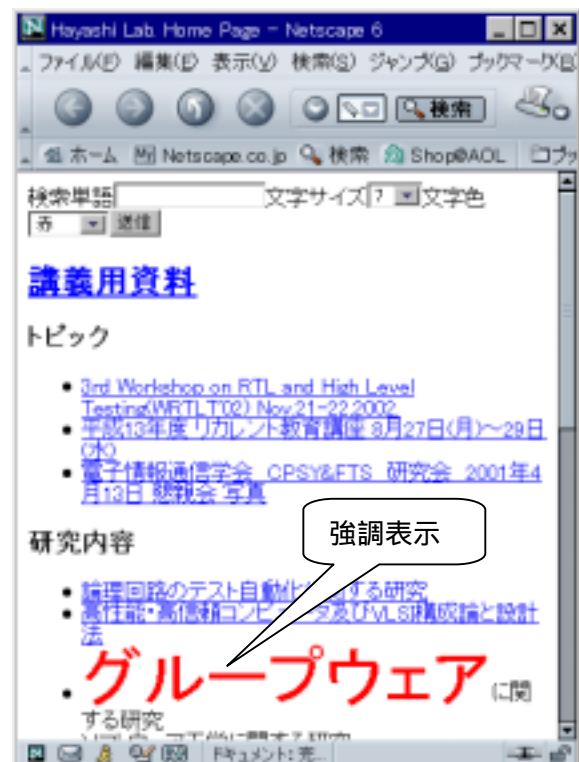


図 7: キーワード強調表示サービス

6.2. 読み込み時間表示サービスと複数のサービスの利用

Web ページ内のリンク先の読み込み時間を時計型のアイコンで表現するサービスと先に示したキーワード強調表示サービスを同時に利用した例を図 8 に示す。読み込み時間は Proxy の持つアクセスログから求める。読み込み時間が表示されることで、表示に時間のかかる Web ページ閲覧するかどうかを判断するための情報が得られる。

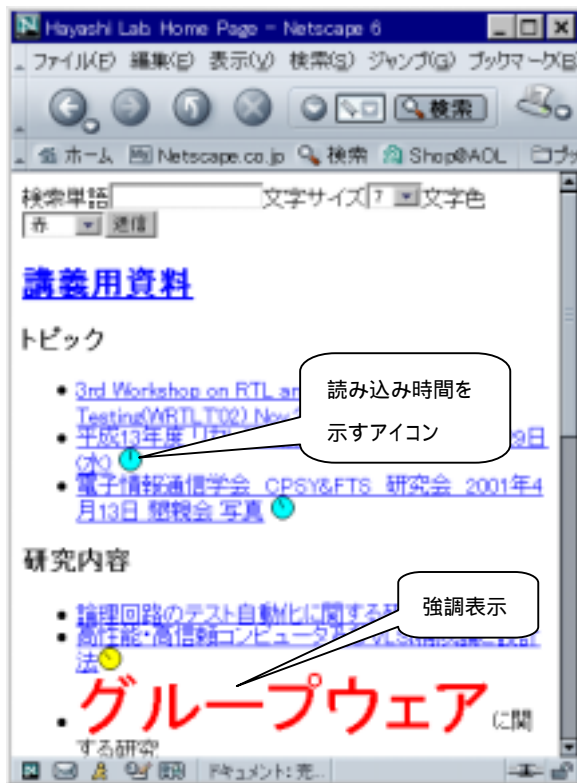


図 8: 読み込み時間表示とキーワード強調サービス

7. 考察

7.1. 開発コスト

前章で紹介したサービスは、いずれも 100 行程度の小さなサイズのプログラムで実現できている。プラットフォームを用いない場合は、HTTP 要求・応答の中継機能などを作らなければならない、プログラムサイズは 1000 行を越えていたと考えられる。このように、提案したプラットフォームを利用することで、サービス製作コストは下がる。

7.2. 複数のサービスの利用

提案したプラットフォームでは、利用者は Web ブラウザの Proxy についての設定を行い、利用者認証のために名前とパスワードを入力し、自分の利用するサービスを選択しなければならない。

Web ブラウザの設定の Proxy についての項目は、一度設定すればその次からは行う必要がないので利用者の負担にはならない。名前とパスワードの入力は、一度入力すると Web ブラウザが記憶してくれるため、Web ブラウザを起動したときに一度入力するだけでよい。利用するサービスの選択は、プラットフォームのサービス選択機能が提供する Web インタフェースで行う。利用したいサービスを選択し、ボタンを押すだけなので、操作は簡単である。

このように、利用者は複数のサービスを簡単に利用することができる。

8. まとめ

本研究では、一つの Proxy で複数の Web サービスを提供できるプラットフォームの提案を行った。また、提案したプラットフォームを用いて Web サービスを試作することにより、プラットフォームとしての有効性を示した。提案したプラットフォームの導入により、利用者にとっては複数の Web サービスの利用が可能になり、サービスの開発者にとってはサービスの開発が容易になる。

参考文献

- [1] 服部, 北, 石田, 沖廣, 加藤, 林: 新たなサービスの追加が容易な Web ページフィルタリングシステムの提案, 情報処理学会研究報告 2001-GW-38, pp.19-24, 2001.
- [2] 村瀬茂樹, 北英彦, 林照峯: Proxy サーバを用いた新たなサービスの可能性について, 情報処理学会研究報告 2000-GW-34, pp.19-24, 2000.