

# 商用 WWW サービスの IPv6 環境移行技術の研究

河合 栄治\* 白波瀬 章† 塚田 清志‡ 山口 英§

## 概要

費用対効果, サービス品質, セキュリティなどが重要となる商用 WWW サービスの IPv6 への移行において, 単に IPv4 および IPv6 用のシステムを個別に用意したり, デュアルスタック構成にする方法では, IPv6 未対応のサービスに関するコンテンツ管理コストの問題や, IPv6 ネットワークセキュリティの問題などが発生する. 本研究では, これらの問題を解決する IPv6 対応のリバースプロキシサーバを開発した. 本システムで主に開発したのは, 高速メモリキャッシュによる処理能力の改善, IPv6-IPv4 プロキシ機能, IPv6 非対応サービスを含むバックエンドサーバに対するアクセスの適切な取り扱いである. 本技術により, 商用 WWW サービスの IPv6 への段階的な移行が可能となる.

キーワード: IPv6, WWW, リバースプロキシサーバ, IPv6 移行技術, 商用サービス

## A Study of IPv6 Transition of Enterprise WWW Services

Eiji Kawai\* Akira Shirahase† Kiyoshi Tsukada‡ Suguru Yamaguchi§

### Abstract

For IPv6 transition of enterprise WWW services, just making a WWW server ready for IPv6 is a weak solution. It does not provide a cost-effective method to handle errors caused by services that have not yet supported IPv6. In addition, introduction of IPv6 hosts to an organization network can break the present IPv4 network security, protected usually by a firewall, because many organizations have not prepared against malicious IPv6 accesses. To address these issues, we developed a high performance IPv6-enabled reverse proxy server that manages IPv6-not-ready services intensively. Our reverse proxy also provides good conformity to the present IPv4 security framework.

**keywords:** IPv6, WWW, reverse proxy server, IPv6 transition, enterprise service

## 1 はじめに

現在, インターネットは数多くの既存通信基盤を統合する次世代通信インフラストラクチャとして認識されるようになり, いわゆるパーソナルコンピュータだけでなく, 携帯電話や PDA などの情報端末, 冷蔵庫, 電子レンジといった家電製品などもインターネットに接続されるようになりつつある. そのため, インターネットに接続される機器の数が急激に増加し, 現在主に利用されているインターネットプロト

コルである IPv4 において, アドレス枯渇問題をはじめとする様々な問題が表面化してきた.

一方で, この問題については早くから指摘されており, アドレス空間を拡張した IPv6 が開発されている. IPv6 は, その拡張により, 実用上は無限に近い多数のアドレス割り当てを可能にし, さらにはモバイル性のサポートなど数多くの機能を有している. また, IPv6 は IPv4 との共存を考慮した設計がなされている [1] ため, 小規模ネットワークでは比較的容易に IPv6 への全面移行が可能であり, 実例も報告されている [2].

一方で, IPv6 への移行を, 範囲が限定された小規模ネットワークではなく, インターネット全体で実現しようとする場合, 現在インターネット利用の多くを占めると考えられている商用 WWW サービスにおいて移行を実現することは非常に重要である. しかし, それは単に WWW サーバを IPv6 化するだけでは実現しない. WWW サービス特有の現実的な問題があるからである. 具体的には, IPv6 未対応の

\* 科学技術振興事業団 さきがけ研究 21  
PREST, Japan Science and Technology Corporation  
eiji-ka@is.aist-nara.ac.jp

† エヌ・ティ・ティ・スマートコネク ト株式会社  
NTT SmartConnect Corp.  
akira@ntt-smc.com

‡ 株式会社 毎日放送  
Mainichi Broadcasting System, Inc.  
tsukada@mbs.co.jp

§ 奈良先端科学技術大学院大学 情報科学研究科  
Graduate School of Information Science, Nara Institute of  
Science and Technology  
suguru@is.aist-nara.ac.jp

サービスへのリンクにおいてエラーが発生することが挙げられる。また、IPv6ホストの安易な導入により、通常ファイアウォールなどによって守られることが多いIPv4ネットワークのセキュリティを損なう恐れがある。商用WWWサービスでは、費用対効果、サービス品質、セキュリティについて特に配慮が必要である。そのため、これらの問題点は許容できず、解決しなければならない。

本研究では、商用WWWサービスのIPv6への移行を実現するために、リバースプロキシサーバによるサーバサイドキャッシュ技術を用いて、システム変更を最小限に抑さえつつ既存のIPv4環境のIPv6への対応を実現することを目的としている。本システムにより、従来のIPv4向けサービスの品質およびセキュリティ保護のフレームワークを維持したまま、同時にIPv6環境へのサービスも低コストで行うことが可能となる。また、一部のIPv6未対応なサービスに対しても、リバースプロキシサーバにおいて一元的に管理し、エラーが発生しないようにすることができる。

以下、本論文の本章以降の構成を示す。まず2章では、WWWサービスのIPv6移行に関する問題点と主なIPv6移行技術について議論する。3章では、本研究で開発したリバースプロキシサーバの設計および実装について述べる。本プロキシサーバは、ベンチマークテストによる性能評価および実際のIPv6ネットワークにおける実証実験を行っており、4章でそれらについて述べる。最後に5章で本研究についてまとめる。

## 2 WWWサービスのIPv6移行における問題点と解決手法

本章では、まずIPv6への移行におけるWWWサービス特有の問題点について、次に商用WWWサービスのIPv6移行技術が満足すべき条件について議論する。さらに、そこでの議論を踏まえ、具体的なIPv6移行手法を分類し、それらの利点および欠点について考察する。

### 2.1 WWWサービスのIPv6移行における問題点

WWWサービスのIPv6への移行における問題点の一つとして、WWWサーバをIPv6に対応させても、場合によってはエラーが発生することが挙げられる。WWWはCGIをはじめとする各種アプリケーションをサービスする目的でも使われ、これらがIPv6に対応していなければエラーが発生する。

さらに、WWWサービスの本質は様々なサービスを互いにリンクすることであることから、WWWはHTTPによるデータの転送だけでなく、他のHTTP以外のプロトコルを用いたサービスのインタフェースとしても用いられている。そのため、IPv6に未対応なサービスをリンクしている場合、ユーザがアクセスすればエラーが発生する。

次に、IPv6ホストの安易な導入はセキュリティの低下を招くという問題が挙げられる。現在IPv6は移行期に入ったばかりであり、長く実験ネットワークとして運用されていたことから、多くのサイトでIPv6ネットワークのセキュリティについて考慮されていない。また、高速ネットワーク向けIPv6ファイアウォールなどの包括的なセキュリティ装置が未だ存在しないため、総合的な対策を取ることが難しい。IPv6では、ファイアウォールなどの中継装置ではなく、各ホストでセキュリティを保護するべきであるという考え方もある<sup>1</sup>ため、さらに問題は複雑である。

### 2.2 商用WWWサービスのIPv6への移行実現の条件

前節では、WWWサービスをIPv6へ移行する際の技術的な問題点について検討した。本節では、さらに、商用WWWサービスのIPv6環境移行策が満たすべき条件について考察する。これらの条件を満たさないIPv6環境移行策は、たとえ技術的問題が無くても、多くの場合採用することができない。

まず、商用WWWサービスのIPv6への移行期においては、現行のIPv4サービスから隔離したIPv6サービスが提供できなければならない。IPv4サービスからの隔離とは、論理的だけではなく、物理的にも隔離したIPv6サービスシステムが必要であることを意味している。これにより、万が一IPv6サービスに障害が発生したとしても、現行のIPv4サービスには一切影響を与えずに障害の分析および復旧にあたることが可能となる。現在のWWWサービスは様々な種類の情報を提供するプラットフォームとして利用されており、そのサービス品質は、サービス提供者(法人や組織)の評価に直接結び付く重要なものと認識されていることから、このことは非常に重要である。

二つ目の条件は、提供されるIPv6サービスは低コストで高い性能を達成しなければならないことである。商用WWWサービスには、非常に多くのリクエストが集中するため、高い性能が求められる。一方で、コスト意識も非常に強く、IPv6サービスの導

<sup>1</sup>セキュリティ保護に関する方針の一つであり、IPv6がファイアウォールを排除しているわけではない。

入に高いコストが強いられるようでは移行の実現は困難である。このコストには、サーバコスト、ネットワークコスト、各種管理コスト、セキュリティコストなどが含まれる。

## 2.3 WWW サービスの IPv6 移行手法

本節では、WWW サービスを IPv6 対応させる具体的な手法について考察する。ここでは、IPv4/IPv6 デュアルスタックサーバを用いる手法、IPv6 専用のサーバを用いる手法、そして本研究の手法である IPv6-IPv4 リバースプロキシサーバを用いる手法に分類する。また、それぞれについて利点および欠点について考察する。

### 2.3.1 IPv4/IPv6 デュアルスタックサーバ

IPv6 をサポートするためには、まず第一にオペレーティングシステムおよびサーバアプリケーションの両方で IPv6 に対応する必要がある。現在、IPv6 は Linux や各種 BSD, Solaris, Windows 2000/XP など多くのオペレーティングシステムでサポートされており、さらには、最もよく用いられている WWW サーバである Apache においても、IPv6 はサポートされている。そのため、現在用いているサーバホストのカーネルおよび WWW サーバを IPv6 対応のものに変更し、IPv6 ネットワークに接続すれば、IPv6 クライアントからのリクエストを処理することが可能となる。

一方で、本手法は先に挙げた IPv6 サービスの隔離という観点からは問題がある。まず、障害発生時にその影響が IPv4 を含めた WWW サービス全体に及ぶ可能性がある。また、同一ホストで IPv6 および IPv4 の両方の処理を行うため、IPv6 のリクエストが集中した場合の IPv4 サービスに対する影響も無視できない。さらには、IPv4 で負荷分散のためのサーバクラスタを構築している場合、IPv6 サーバクラスタ技術が未発達であるため問題となる。

次に、IPv6 未対応のサービスを管理するために、仮想ホスト機能などを利用し、IPv6 用にコンテンツを複製する必要があるのも問題である。そのため、IPv6 用のコンテンツ管理に加え、コンテンツ更新時に同期を取らなければならないなど、管理コストが高い。このように、本手法は最も手軽であるが、問題が多く、現実的な解決策にはならない。

### 2.3.2 IPv6 専用のサーバ

IPv6 専用ホストを別途用意し、システムを複製することで IPv6 サービスを構築することができる。本手法の利点は IPv6 サービスの隔離が実現できることであり、IPv6 サービスへの負荷の集中や故障、

攻撃などから IPv4 サービスを保護することができる。しかしながら、デュアルスタックサーバと同様に、システムの複製によるコンテンツ管理コストが大きいという問題は未解決のままである。

### 2.3.3 IPv6-IPv4 リバースプロキシサーバ

IPv6 のリクエストを受信し、IPv4 のリクエストに変換して IPv4 サーバに中継するリバースプロキシサーバ(サーバアクセラレータとも呼ばれる)を別途用意することにより、IPv6 サービスを実現することができる。この手法は、様々な利点を持っている。

まず、IPv6 専用のサーバと同様、IPv6 サービスの隔離を実現することができる。次に、コンテンツ管理コストを削減することができる。これは、プロキシサーバが自律的にオリジンサーバのコンテンツを取得するからである。障害からの復旧においても、コンテンツの同期に関しては自動化されていると考えてよい。さらに、エラーが発生する IPv6 未対応サービスへのリンクなどもプロキシサーバで一元管理することができる。最後に、負荷の集中という問題に対しても、プロキシサーバ上でキャッシュ技術を用いることによって、高い性能を達成することができることも利点の一つである。一般的にリバースプロキシサーバは高いヒット率を達成することが可能であり [3]、既存 IPv4 サービスへの影響を小さくすることができる。これは、高いヒット率により、オリジンサーバへ転送されるリクエストを少なくすることができるからである。

## 3 設計と実装

本研究では、商用 WWW サービスの IPv6 への移行における諸問題を解決するため、リバースプロキシサーバによる IPv6-IPv4 中継システムを開発した。本章ではその設計と実装について述べる。

### 3.1 設計

本システムの主な技術開発点は、リバースプロキシサーバにおける IPv6-IPv4 中継、高性能メモリキャッシュによる処理能力の改善、サービスフィルタによる柔軟かつ集約的なコンテンツ管理の三点である。本節では、それらの設計について述べる。

#### 3.1.1 リバースプロキシサーバにおける IPv6-IPv4 中継

リバースプロキシサーバにおいて IPv6-IPv4 中継を行うことによって、IPv4 用 WWW サーバを用いた IPv6 環境への WWW サービスの提供を可能にする。クライアントのアクセスに対する具体的な処理

手順は以下の通りである。外部からの IPv6 のアクセスに対して、可能な場合はプロキシサーバが保持しているキャッシュから直接コンテンツを返信する。また、オリジンサーバへのアクセスが必要な場合には、プロキシサーバは IPv4 を用いてオリジンサーバからコンテンツを一旦受け取り、キャッシュに格納しながら、IPv6 を用いてクライアントに転送する。この手順により、IPv6 環境からの IPv4 WWW サービスへの透過的なアクセス手段を実現する。本手法は、2.3 節で述べた様々な利点を持っている。

### 3.1.2 高性能メモリキャッシュ

近年、WWW サーバにおける高い性能を達成するために、メモリベースサーバが開発されている [4]。メモリベースサーバとは、ファイル I/O を削減するために、サーバプロセス用のロックされた (ページアウトしない) メモリ領域を確保しファイルキャッシュとして用いるものである。本研究で開発するリバースプロキシサーバもメモリベースであり、特にメモリキャッシュの効率を高めるキャッシュアルゴリズムを開発した。

リバースプロキシサーバでは、いわゆるユーザサイドプロキシサーバ (フォワードプロキシサーバとも呼ばれる) と比較して、対象となるサービスコンテンツおよびその容量が限定される。このことから、キャッシュ容量が十分用意されたリバースプロキシサーバでは、非常に高いキャッシュヒット率が達成可能である。一方で、現在一般的に利用可能なメモリ容量と単一の商用 WWW サーバで提供されるコンテンツの総容量を比較すると、ほとんどの場合メモリ容量の方が小さい。さらに、搭載されているメモリの全てをオブジェクトのキャッシュとして用いることができるわけではない。なぜなら、オペレーティングシステム、バッファキャッシュ、他のプロセス、さらにはプロキシサーバ自身のバッファなどにもメモリは消費されるからである。このことから、キャッシュに利用可能なメモリ容量は、コンテンツ総容量の 10% 以下であることが多い。そのため、効率的なメモリ管理が非常に重要となる。

実装が容易であることからキャッシュアルゴリズムとしてよく用いられている LRU (Least Recently Used Algorithm) は、アクセス回数が多いコンテンツは短期間に繰り返しアクセスされるという仮定のもと、キャッシュを管理する。しかし、プロキシが提供するコンテンツの総容量と比較してキャッシュ容量が小さい場合、非常にアクセス周期の短いごく一部のオブジェクトしかキャッシュヒットせず、キャッシュを有効に利用することができない。

一方、LFU (Least Frequently Used Algorithm) は、

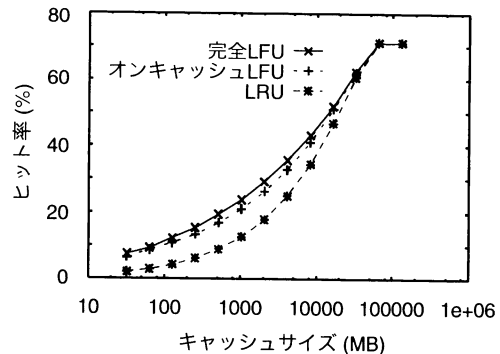


図 1: LRU と LFU のシミュレーション結果

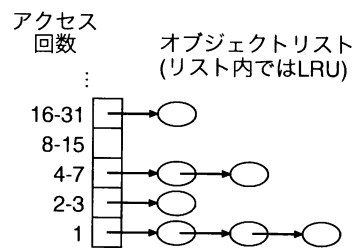


図 2: 疑似 LFU のデータ構造

各オブジェクトをアクセス回数でソートしておく必要があるため、LRU と比較して実装が複雑であるが、特にキャッシュ容量が小さい場合に良好なキャッシュヒット率を示すことがシミュレーションの結果判明した (図 1)。このシミュレーションでは、リクエスト分布には Zipf の法則、オブジェクトサイズは対数正規分布を用い [5]、1,300 万のオブジェクトに対して 3,300 万のリクエストを生成した。ここで、完全 LFU とは、置き換えによりキャッシュから削除されたオブジェクトを含む全てのオブジェクトについてアクセスカウントを保持するアルゴリズムであり、アクセスカウント用の記憶領域のコストを無視しているが、LFU としては最良の結果となる。また、オンキャッシュ LFU とは、キャッシュされているオブジェクトに関してのみアクセスカウントを保持するものであり、実現的な実装と考えられる。

一方で、LFU をリバースプロキシサーバに用いる場合の問題点は、ワーキングセットの変化に対応できないことである。これは、LFU がアクセス回数のみを考慮するために発生する。

以上の考察から、本実装では LFU を基礎として、オブジェクトのエージング処理を行うことにより、効率的なキャッシュ管理を実現した。基本的なデータ構造としては、図 2 に示したように、リンクリストの配列を用いることにより、擬似的な LFU を実

現した。オブジェクトのエージングは、ガーベージコレクションを行う際に、各オブジェクトの最終アクセス時刻に、そのオブジェクトのアクセス回数によって線形に増加するオブジェクト保持時間を加えた時刻と、現時刻を比較することによって実現しており、C言語による疑似コードは以下の通りである。

```
for(;;){
    obj = gc_next_obj();
    if(obj->last_access + hold_min
        + hold_inc * obj->n_access < now)
        gc(obj);
}
```

これにより、アクセス回数の多いオブジェクトは周期が多少長くてもキャッシュに残りやすくなる。また一方で、たとえアクセス回数が多くても、一定時間アクセスされなくなったオブジェクトは、キャッシュから除去されるようになる。

### 3.1.3 サービスフィルタ

現在の WWW サービスは、いわゆる HTTP を用いたサーバとクライアント間の通信だけで行われているわけではなく、ストリーミングサービスなどの HTTP 以外のプロトコルを用いたサービス<sup>2</sup>も WWW を通じて提供されている。このようなサービスは、リバースプロキシサーバによる HTTP 通信の中継だけでは IPv6 に対応できない。サーバおよびクライアントを含むサービス全体を IPv6 に対応する必要があるからである。しかし、現在様々なサービスが IPv6 への対応を進めている段階であり、IPv6 対応が完了しているサービスはまだ少数にとどまっている。このような状況で WWW だけ IPv6 に対応しても、IPv6 に対応していないサービスにおいてクライアントでエラーが発生する結果となり、商用 WWW サービスでは許容できない。

そこで、本研究では、IPv6 に対応していないサービスのリストをプロキシサーバにおいて一括管理し、それらについてはクライアントに対して IPv6 に未対応のサービスであるという内容のメッセージを送信する機構を開発した。これにより、クライアントで発生するエラーを防ぐことができる。具体的には、IPv6 未対応のサービスに対するフィルタを用意し、そのようなサービスに対するリクエストを捕捉し、ユーザへの説明を表示するページにリクエストを転送するようクライアントに通知する。フィルタには、削除したいサービスの URL をファイル拡張子もしくはその完全パス名で指定する。また、削

<sup>2</sup>HTTP 以外の通信を遮断しているファイアウォール環境のために、HTTP を用いるストリーミングサービスも存在する。

除した URL のそれぞれについて、未対応であるというメッセージを伝える代替ページの URL を指定することができる。ユーザへの通知には、HTTP 応答におけるステータスコードである “301 Moved Temporarily” を用いた。

## 3.2 実装

これまでに筆者らは、全国高校野球選手権大会のインターネット中継実験において、新しく開発したりバースプロキシサーバ Chamomile を用いた WWW システムを運用し、多くの知見を得てきた [6]。本システムは、その Chamomile を基に 3.1 節で述べた各種機能を追加している。

Chamomile の実装は、pthread ライブラリを用いたマルチスレッドアーキテクチャを採用し、性能を重視した設計になっている。特に、SMP ホスト上で運用した場合に性能のスケラビリティを確保するよう、各種 thread の生成数はホストの構成、負荷の状況に応じて柔軟に設定可能となっている。一方で、移植性にも配慮した実装を行っており、現在 Linux、FreeBSD、Solaris 上での動作を確認している。

## 4 運用および評価

Chamomile の基本的な性能評価として、プロキシベンチマークによる実験を行った。また、本システムを実際の IPv6 ネットワーク上で運用し、実証実験を行った。本章では、それらの結果についてまとめる。

### 4.1 ベンチマークテスト

まず、本研究で開発したりバースプロキシサーバの基本性能を検証するために、2001 年 10 月に行われた第 4 回 Cache-Off [7] に参加した。Cache-Off とは、キャッシュプロキシサーバを開発している企業および研究者が一堂に会し、それぞれの製品の性能をベンチマークソフトウェア WebPolygraph [8] を用いて計測するものである。今回の参加では、PC (CPU: Pentium II 450MHz, メモリ: 1GB, OS: Linux 2.4) 上で、毎秒 1,000 リクエストの処理能力を達成した。また、ヒット率も約 70% と十分高い値であった。これらの値は、多くの商用 WWW サービスとして十分な性能である。

### 4.2 IPv6 ネットワークにおける実運用

次に、IPv6 ネットワークにおける実運用による実証実験を行うために、2001 年 12 月に行われたイベント Net.Liferium2001 において本システムを運用

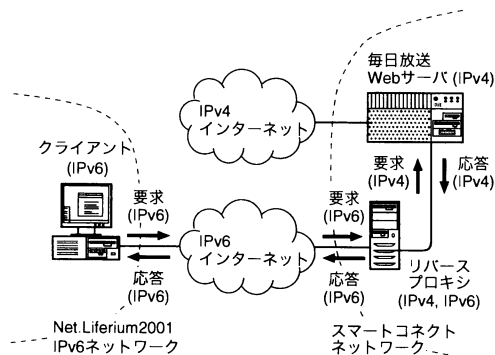


図 3: 実証実験ネットワーク

した。ネットワーク環境の概略を、図 3 に示す。ここでは、Net.Liferium2001 の IPv6 ネットワーク上に設置したホストから、Chamomile 経由で毎日放送の WWW サイトにアクセスする環境を用意した。本実験では、IPv4 サービスが存在しない IPv6 のみの環境に置かれたクライアントから IPv4 サーバが正しくアクセスでき、開発したフィルタ機能などが正しく動作していることを検証した。

## 5 まとめ

現在、多くのオペレーティングシステムおよびサーバプログラムで IPv6 がサポートされ、単に IPv6 による WWW サービスを行うのは比較的容易である。しかし、費用対効果、サービス品質、およびセキュリティが重要視される商用 WWW サービスでは、単純な解決策では多くの問題が発生する。本研究では、これらの問題点を検討した結果、リバースプロキシサーバを用いたシステムが現時点では最良の解決策であるという結論に達し、高性能かつ柔軟なコンテンツ管理を可能にするリバースプロキシサーバを開発した。

本システムで主に開発したのは、リバースプロキシサーバにおける IPv6-IPv4 中継、高性能メモリキャッシュ、柔軟かつ集約的なコンテンツ管理を実現するサービスフィルタの三点であった。これらの技術を検証するために、本研究では第 4 回 Cache-Off におけるベンチマーク評価、さらには Net.Liferium2001 における IPv6 ネットワーク上での実運用を行った。それにより、本システムが、既存の商用 WWW サービスにおける IPv6 への移行を実現する有効な解決策となることを実証した。

## 謝辞

本研究における運用実験で御協力頂いたサイバー関西プロジェクト<sup>3</sup> (CKP) に感謝致します。

## 参考文献

- [1] R. Gilligan and E. Nordmark. Transition Mechanisms for IPv6 Hosts and Routers. RFC 2893, August 2000. <http://www.ietf.org/rfc/rfc2893.txt>.
- [2] Masahiko Kimoto, Hiroki Suenaga, Toshihiro Kimura, and Hiroyuki Ohno. Construction of IPv6 Small Network Using INS router. In *Proceedings of Inet 2000*, Yokohama, Japan, July 2000.
- [3] Ari Luotonen. *Web Proxy Servers*. Prentice-Hall, 1998.
- [4] Philippe Joubert, Rovert B. King, Rich Neves, Mark Russinovich, and John M. Tracy. High-Performance Memory-Based Web Servers: Kernel and User-Space Performance. In *Proceedings of the 2001 USENIX Annual Technical Conference*, Boston, Massachusetts, USA, June 2001.
- [5] Paul Barford and Mark Crovella. Generating Representative Web Workloads for Network and Server Performance Evaluation. In *Proceedings of ACM SIGMETRICS '98*, June 1998.
- [6] 西馬一郎, 河合栄治, 知念賢一, 吉田豊一, 香取啓志, 山口英. WWW クラスタにおける同期を考慮したコンテンツ更新機構の設計と実相—夏の高校野球 WWW 中継における運用—. インターネットコンファレンス, 2001 年 11 月.
- [7] 4th TMF Cache-Off, November 2001. <http://www.measurement-factory.com/result/public/cacheoff/N04/>.
- [8] Web Polygraph. <http://www.web-polygraph.org/>.

<sup>3</sup><http://www.ckp.jp/>