

# HTTP コネクションの維持による利用終了検知を行う新たな Opengate の導入とその運用

大谷 誠<sup>†</sup>, 江藤 博文<sup>†</sup>, 渡辺 健次<sup>‡</sup>, 只木 進一<sup>†</sup>, 渡辺 義明<sup>‡</sup>

<sup>†</sup> 佐賀大学 総合情報基盤センター

<sup>‡</sup> 佐賀大学 理工学部

## 概要:

佐賀大学では, 利用者端末や公開端末からのネットワーク利用を認証・記録する Opengate を開発・公開し, 学内で運用を行ってきた. この Opengate は, ネットワークの利用終了の即時検知に Java Applet を用いるため, Java の実行環境を持たない端末における利用終了の即時検知には対応していなかった. そこで, このような端末に対応するため, HTTP の遅延応答と HTTP Keep-Alive 機能を用いた利用終了の即時検知機能を実装した新たな Opengate を開発し, 運用を行った. 本稿では, HTTP Keep-Alive 機能による終了検知を行う新しいネットワーク利用者認証システム Opengate の概要と, その導入・運用について述べる.

## Installation and Operation of New Opengate with Detecting Usage Termination by Watching HTTP Connection

Makoto Otani<sup>†</sup>, Hirofumi Eto<sup>†</sup>, Kenzi Watanabe<sup>‡</sup>,  
Shin-ichi Tadaki<sup>†</sup>, Yoshiaki Watanabe<sup>‡</sup>

<sup>†</sup> Computer and Network Center, Saga University

<sup>‡</sup> Faculty of Science and Engineering, Saga University

## Abstract:

We have developed and distributed a network user authentication system “Opengate”. After the authentication, the system sends Java Applet to the terminal and watches the usage. Therefore, on a terminal without Java plug-in, usage termination is not detected immediately. We developed new Opengate which solves this problem by using delayed HTTP response and HTTP Keep-Alive. And the service of new Opengate was started at our university. This paper describes installation and management of the new Opengate capable of detecting usage termination by delayed HTTP response and HTTP Keep-Alive function.

## 1 はじめに

佐賀大学では, 利用者端末や公開端末からのネットワーク利用を認証・記録する「Opengate」を開発・公開し, 2001 年より学内においてディスクレスで運用を行っている<sup>1)-3)</sup>. また, 2005 年からは IPv6 に対応した Opengate の試験運用を行っている<sup>4), 5)</sup>. この Opengate は, Web アクセスによって認証画面が提供される平易なインタフェースを持ち, 認証には既設の LDAP や RADIUS

サーバなどを利用することができる.

これまで佐賀大学で運用してきた Opengate は, 利用者のネットワークの利用終了の即時検知に Java Applet を用いる. 開発当初は, 利用者端末の多くに購入当初から Java の実行環境が導入されており, Java Applet による即時検知が有効に機能した. しかし, 近年の利用者端末の多くは, あらかじめ Java の実行環境が導入されていない. このような端末で即時検知を行うためには, Opengate の利用前に Java の実行環境を導入する必要性が生じる. また, これに伴う指導も必要であった. 実

際の運用では、利用の約 24.5% (2007 年 1 月の統計) で、Java Applet による利用終了の即時検知が利用できず、一定時間経過後に遅延閉鎖する状況であった。Opengate の利用記録から判断して、その多くが Java が利用できない Web ブラウザでは無く、Java の実行環境が導入されていない一般的な Web ブラウザであると考えられた。

そこで、追加プラグインや拡張機能を持たない標準的な Web ブラウザにおいても利用可能な HTTP による即時検知方法を考案し、Opengate に実装した<sup>6)</sup>。この新たな Opengate を、学内において 2007 年 4 月より全学規模で運用を開始した。また、Opengate は既に数年にわたり教育研究の現場で利用しているため、新たな Opengate へ円滑に移行が行えるように従来の検知方式に加える形で新たな検知方法を導入した。さらにインターフェースも従来のものを引き継ぐなどの配慮を行った。

本稿では、新たな Opengate の概要と、佐賀大学における導入と運用について述べる。

## 2 Opengate の概要

### 2.1 概要

Opengate は、特定多数の利用者が多様な端末を接続するネットワーク環境において、利用者認証と利用記録を行うことができるシステムである。Opengate が設置されたネットワークでは、Web ブラウザさえあれば、特別な申請やソフトウェアの準備なしに、有資格者のみが利用者端末をインターネットに接続することができる。

Opengate のシステム構成例を図 1 に、ソフトウェアの構成を図 2 に示す。

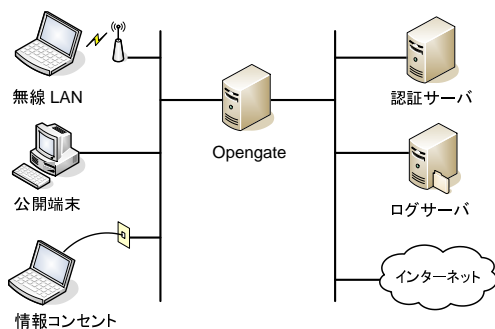


図 1 Opengate のシステム構成例

Opengate は、次のように動作する。利用者がネットワークに利用者端末を接続した後、始めに任意の Web サイトにアクセスする。Opengate はその通信を奪い取り、代わりにネットワーク利用の認証ページを利用者の Web ブラウザに提供する。利用者は、この認証ページに利用者 ID とパスワードを入力する。Opengate は、入力された利用者情報を認証サーバへ問い合わせる。認証が成功

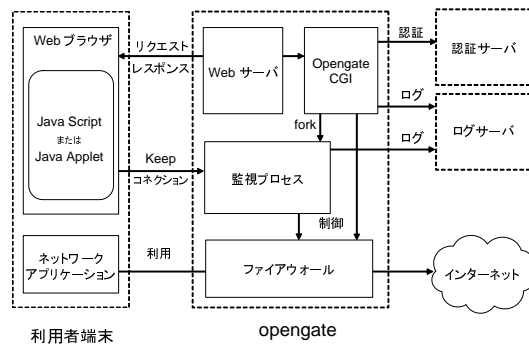


図 2 Opengate のソフトウェア構成

した場合、当該端末の IP アドレスの開放ルールをファイアウォールに加える。

### 2.2 従来の Opengate の動作

従来の Opengate では、認証成功後、認証完了ページとともに Web ブラウザに Java Applet をダウンロードさせる。Opengate は利用者端末ごとの監視プロセスを起動する。Java Applet はこの監視プロセスとの間に、TCP コネクションを維持することによって、ネットワークの利用を監視する。利用者が Web ブラウザを終了、もしくは利用者端末の OS が終了すると、TCP コネクションが切断される。その切断を Opengate の監視プロセスが検知し、ファイアウォールの開放ルールを削除する。

なお、HTTP 以外の通信プロトコルを使用する場合も、まず始めに任意の Web サーバへアクセスし、認証を行わなければならない。

## 3 新たな Opengate の概要

### 3.1 概要

第 2.2 節で述べたように、従来の Opengate では、利用者端末の Web ブラウザに送った Java Applet と監視プロセスとの間の TCP コネクションを監視する。そして、この TCP コネクションの終了によって、利用の終了を即時に検知する仕組みを導入している。

新たな Opengate では、追加プラグインや拡張機能を持たない標準的な Web ブラウザにおいても利用可能な、HTTP の遅延応答 (以下、HTTP 遅延応答) と、HTTP/1.1<sup>7)</sup> において標準となった Keep-Alive 機能 (以下、HTTP Keep-Alive) を用いる利用終了の即時検知 (以下、HTTP による終了検知) を行う。

### 3.2 HTTP による終了検知の動作

HTTP による終了検知の動作を、以下に述べる (図 3)。

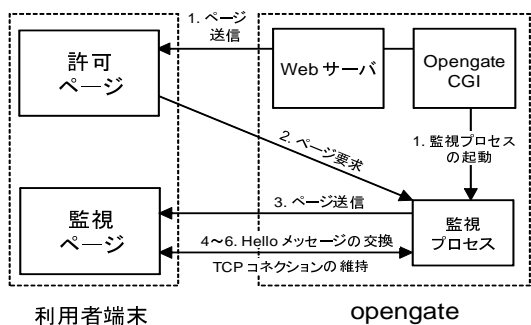


図3 利用監視の動作

1. 認証終了後,Opengate CGIは,許可ページを Web ブラウザに送信するとともに,監視プロセスを起動する.
2. 許可ページ内において JavaScript を実行し,監視プロセスに対して監視ページを要求する.
3. 監視プロセスは,監視ページを Web ブラウザに送信する.
4. 監視ページ内において JavaScript を実行する. JavaScript は,サーバと非同期で HTTP 通信を行うための“XMLHttpRequest”を発行し,監視プロセスに対して“hello メッセージ”を送信する.
5. 監視プロセスは,XMLHttpRequest による hello メッセージに対して,一定時間(標準設定で 100 秒)遅延させ,返答する.
6. Web ブラウザは,hello メッセージに対する返答を監視プロセスから受け取ったら,すぐに次のXMLHttpRequest を発行し,hello メッセージを送信する.以降は(4)のhello メッセージ送信からの処理を繰り返す.

上記の(4)から(6)の繰り返しの様子を図4に示す.

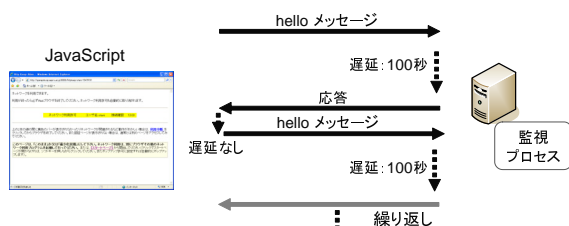


図4 HTTP による接続維持

XMLHttpRequest から応答 (HTTP 遅延応答) までの間は, Web ブラウザの標準的な機能によって TCP コネクションは維持される.(6)における返答を受けて再度XMLHttpRequest を発行する間も,HTTP Keep-Alive によって TCP コネクションが維持されると期待できる.

監視プロセスは,この間 TCP コネクションを監視し続ける.この TCP コネクションが切断されたら,これ

を検知し,利用者がネットワーク利用を終了したと判断する.

### 3.3 従来の終了検知方法の併用

第 3.2 節で述べた方法で,一般的に利用されている多くの Web ブラウザで,HTTP による終了検知が可能であると考えられる.

しかし,新たな Opengate を実際に大学において運用することを考えると,更に多種多様な Web ブラウザが利用されることを想定しなければならない.特に,HTTP による終了検知で利用している HTTP Keep-Alive は HTTP/1.1 から標準となっている.このため HTTP Keep-Alive が実装されていない古い Web ブラウザが利用されることも考えられる.

つまり,HTTP による終了検知を標準的な検知方法としながら,HTTP による終了検知が利用できない Web ブラウザも利用されることも想定する.そのため,従来利用していた終了検知方法など,以下の 6 つの方法を組み合わせて利用することで,様々な Web ブラウザに対応した.

- HTTP-CLOSED

HTTP が維持する TCP コネクションの切断を検知する方法である.これは,上述した HTTP による終了検知方法である.

- JAVA-CLOSED

クライアントに送付した Java Applet との間に維持する TCP コネクションの切断を検知する方法である.これは,従来の Opengate の検知方法である.

- MAC-CHANGED

MAC アドレスが変更されたことを検知する方法である.

- NO-PACKET

利用端末が送受するパケットが長期にわたって無いことを検知する方法である.

- TIME-EXCEEDED

利用者のあらかじめ入力した利用時間が経過したことをチェックする方式である.ただし設定限度(標準設定で 60 分)を設けて,一時的利用に限定する.

- LINK-CLICKED

利用者が利用終了のリンクをクリックしたことをチェックする方式である.

新たな Opengate は,HTTP-CLOSED による検知が主である HTTP 監視モード, JAVA-CLOSED が主である JAVA 監視モード,それらが利用できないときの BASIC 監視モードの 3 つの監視モードを持つ.そして,

以下のようにこれらのモードを選択する。

JavaScript が実行可能で、XMLHttpRequest の発行も可能な場合は、HTTP 監視モードが自動的に選択される。HTTP 監視モードでは、HTTP-CLOSED、MAC-CHANGED が検出された場合はネットワークを閉鎖し終了する。

HTTP 監視モードが利用できず、Java Applet の実行が可能な場合は、JAVA 監視モードが自動的に選択される。JAVA 監視モードでは、JAVA-CLOSED、MAC-CHANGED が検出された場合はネットワークを閉鎖し終了する。

HTTP 監視モードと JAVA 監視モードが選択できない場合や、あらかじめ利用時間が入力されていた場合は、BASIC 監視モードとして動作する。BASIC 監視モードでは、MAC-CHANGED、NO-PACKET、TIME-EXCEEDED、LINK-CLICKED のいずれかを検出するとネットワークを閉鎖し終了する。

## 4 新たな Opengate の導入と運用

### 4.1 導入

佐賀大学では、全学規模で安定かつ低運用コストでサービスを行うために、若干の設定だけが異なる多数の Opengate をディスクレスで運用する仕組みを従来より導入している。これまで、ソフトウェアのバグも無く安定的にサービスを行っている<sup>3)</sup>。

2007 年 4 月 6 日に、従来の Opengate から新たな Opengate に移行し、22 台の Opengate によるディスクレス運用を開始した。移行の際に、ディスクレス環境の再起動などによる 10 分程度のサービス停止が必要となった、しかし、停止を事前にアナウンスしていたこともあり、円滑に導入作業を行うことができた。導入後、利用トラブルや利用の問い合わせも特になく、これまで Opengate の不具合によるサービスの停止も起きていない。

新たな Opengate は、利用終了の検知方法が異なるものの、インタフェースやその利用方法は、従来の Opengate と同じになるよう開発した。よって、利用者は新たな Opengate の導入後も、利用終了の検知方法の違いを意識せず、従来の Opengate と同じように利用できる。このため、導入に伴う新たな利用指導も特に必要なく、使い方の問い合わせなども、新たに発生しなかった。

### 4.2 利用状況

Opengate の利用対象者は、佐賀大学の構成員である学生(約 7,500 人)、教職員(約 1,500 名)、および学外の一時的利用者である。これまでの運用では、月平均で約 2

~3 万回の利用があり、多い時には、約 200 人前後が同時に利用している。

新たな Opengate を導入後、約 3 ヶ月間(2007 年 5 月~7 月)の利用者は 2,511 人で、利用回数は、のべ 73,261 回であった。Opengate の 1 回の平均利用時間は、約 4,412 秒(中央値: 約 2,946 秒)であり、利用者のうち 2,231 人(88.85%)が学生であった(表 1)。

表 1 Opengate の利用者

利用者	利用者数	利用回数
学生	2,231 (88.85%)	65,029 (88.76%)
教職員	272 (10.83%)	7,963 (10.87%)
学外一時利用者	18 (0.72%)	269 (0.37%)

図 5 に、1 日の利用回数のヒストグラムを示す。平日には約 1,000 回前後の利用があり、平日以外でも約 200 回前後の利用がある。

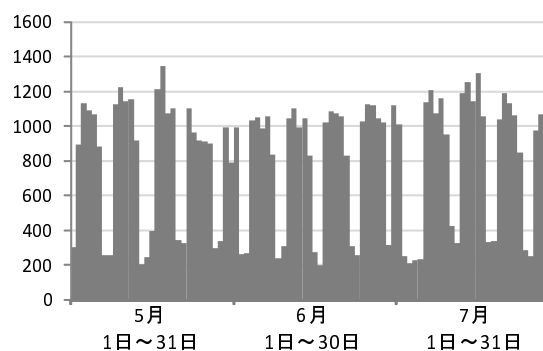


図 5 利用回数のヒストグラム(1日)

設置場所別の利用状況を表 2 に示す。表 2 の場所に Opengate を設置し、その周辺にサービスを提供している。利用場所として、理工学部と農学部が多いことがわかる。これらの学部は学生が多いためと、授業の際に Opengate を利用しているためと思われる。

## 5 考察

### 5.1 動作検証

新たな Opengate は Web ブラウザが直接制御する HTTP コネクションを利用する。このため独自の TCP コネクションを新規に作って利用する Java による検知よりも、Web ブラウザの挙動の影響を受けやすい。そこで、実際の運用開始に先立ち、現在多く利用されている各種の端末や Web ブラウザで TCP コネクションが実際に長期間維持できるかの動作確認を行った。

HTTP による終了検知が正常に利用できない Web プ

表 2 Opengate の利用状況 (設置場所別)

設置場所	設置台数	のべ利用回数
文化教育学部	4	8,211
経済学部	1	4,062
理工学部	9	33,229
農学部	2	17,434
医学部	1	106
附属図書館	1	4,394
学生会館 (生協)	1	789
遠隔施設	1	2,680
産学官連携推進機構	1	1,949
総合情報基盤センター	1	407

ブラウザとして、Windows CE.NET/Mobile で利用される Pocket Internet Explorer や Internet Explorer Mobile、Palm OS で利用される WebPro、PlayStation Portable (PSP) や Zaurus で利用される NetFront を確認した。これ以外の多くの Web ブラウザで正常に動作することが確認できた。Web ブラウザの世界的な利用シェアを考えると、利用者端末の約 99.8% 以上 (2007 年 1 月)<sup>8)</sup> で HTTP による終了検知が可能であると考えられる。

実運用においても、導入後 約 3 ヶ月間で実際に利用された OS (表 3) や Web ブラウザ (表 4) のほとんどは、HTTP による終了検知が可能な Web ブラウザであった。

表 3 利用された OS の割合

OS	利用回数	割合
Windows Vista	6,712	(9.16 %)
Windows XP	54,380	(74.23 %)
Windows 2000	5,707	(7.79 %)
Windows ME	220	(0.30 %)
Windows 98	1,578	(2.15 %)
Windows Mobile	11	(0.02 %)
Mac OS X	3,535	(4.83 %)
Mac OS 9	798	(1.09 %)
Linux	85	(0.12 %)
FreeBSD	3	(0.00 %)
PlayStation Portable	224	(0.31 %)
その他 / 不明	8	(0.01 %)

また、実運用での利用状況も従来の Opengate とほぼ同様であった。よって、新たな Opengate は正常に動作しているものと思われる。

## 5.2 HTTP による終了検知の有用性

新たな Opengate 導入後の約 3 ヶ月において、即時検知が利用できなかった割合は、約 1.18% (表 5) であった。

表 4 利用された Web ブラウザの割合

Web ブラウザ	利用回数	割合
Internet Explorer 7	12,884	(17.59 %)
Internet Explorer 6	46,352	(63.27 %)
Internet Explorer 5	2,207	(3.01 %)
Firefox 2.0	6,460	(8.82 %)
Firefox 1.5	1,334	(1.82 %)
Firefox 1.0	241	(0.33 %)
Netscape 7	477	(0.65 %)
Safari 3	30	(0.04 %)
Safari 2	1,640	(2.24 %)
Safari 1	964	(1.32 %)
Opera 9	326	(0.44 %)
Opera 8	5	(0.01 %)
その他 / 不明	341	(0.47 %)

従来の Opengate では、約 24.5 % (2007 年 1 月) であったため、即時検知の割合が大きく向上した。なお、即時検知ができない場合は、一定時間後に通信路が閉鎖する。

この向上は、HTTP による利用終了の即時検知が、Web ブラウザの標準的な機能のみで実装されているためである。よって、HTTP による利用終了の即時検知は、Opengate において有効に機能していると考えられる。

表 5 利用監視モードの比率

HTTP	JAVA	BASIC (自動選択)	BASIC (利用者が選択)
56,168 回 (76.67 %)	1,669 回 (2.28 %)	865 回 (1.18 %)	14,559 回 (19.87 %)

ただし、利用者が利用時間を入力することで、HTTP 監視モードを回避している例が 19.87% (表 5) と大きな割合を占めていることも分かる。

運用初期の段階で、利用者を観察してその原因を探った。監視ページを維持せずに Web ブラウザを使ったためにネットワークが閉鎖した経験から、それ以降は利用時間を入力して使用している例が多く存在することが分かった。従来の Java Applet による方法では、監視ページから移動しても、Web ブラウザが終了するまで TCP コネクションが維持できていた。よって監視ページを維持せずに利用していたと思われる。

そこで、認証ページのインタフェースを改め、さらに監視ページから他へ移動したり、認証ページを閉じたりする際は、警告を出すように改良した。小規模なテストを行ったところ、利用時間を入力する例がほぼなくなり、ほとんどの利用で利用終了の即時検知が可能となった。

このため、運用環境において、Web インタフェースを

改良するとともに、BASIC 監視モードの利用制限時間を短縮して HTTP 監視モードへの移行を促すなどの検討を今後行う必要があると考える。

### 5.3 起動速度の高速化

HTTP による終了検知では、Java の実行環境の起動を必要としないため、従来の Opengate と比べ、端末側の起動速度が高速化すると考えられる。運用においても、その起動速度の高速化が確認できた。

HTTP 監視モードと JAVA 監視モードでの Opengate の起動時間のヒストグラム (0.1 秒毎) を図 6、図 7 に示す。

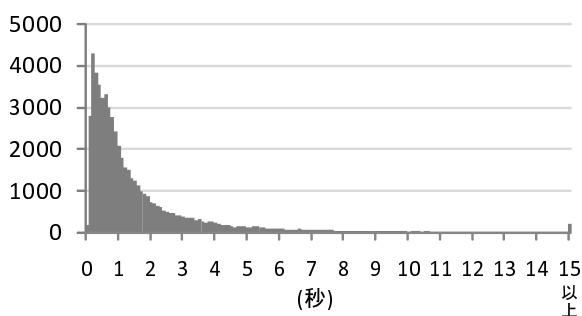


図 6 起動時間 (秒) のヒストグラム (HTTP 監視モード)

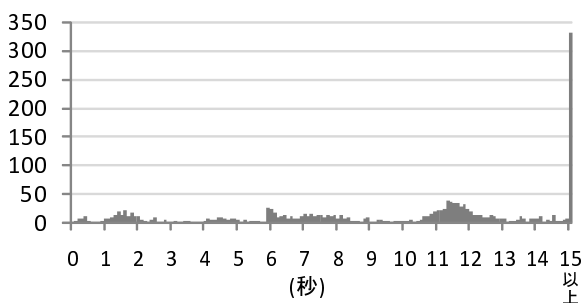


図 7 起動時間 (秒) のヒストグラム (JAVA 監視モード)

新たな Opengate 導入後の約 3 ヶ月において Java Applet を利用した終了検知 (JAVA 監視モード) では、端末側の起動時間が平均で約 12.34 秒であったが、HTTP による終了検知 (HTTP 監視モード) では平均で約 1.77 秒であり、利用者端末での起動時間が大幅に減少した。

## 6 まとめ

本稿では、HTTP 遅延応答と HTTP Keep-Alive による利用終了の即時検知を行う新たな Opengate の概要と、佐賀大学での導入・運用について述べた。

従来の Opengate は、利用終了の即時検知に Java Applet を用いていたため、Java 環境を持たない端末における利用終了の即時検知に対応できなかった。この問題の解決のために、HTTP 遅延応答と HTTP Keep-Alive による利用終了検知機能を実装した新たな Opengate を開発し、2007 年 4 月より運用を開始した。

これにより、Java 環境が導入されていない利用者端末への対応が可能となり、より多くの利用者端末で即時検知が可能となった。また、Java Applet を利用しないため、端末での起動が高速化し、利用者端末の負荷軽減にも繋がった。従来の Opengate の利用方法とインターフェースを引き継いだため、移行も円滑に行うことができた。

## 謝辞

本研究は、平成 17 年度文部省科学研究費補助金 (基盤研究 (C) 課題番号 17500040) の援助を受けている。

## 参考文献

- 1) 渡辺義明 他：「Opengate ホームページ」  
<http://www.cc.saga-u.ac.jp/opengate/>
- 2) 渡辺義明, 渡辺健次, 江藤博文, 只木進一：利用と管理が容易で適用範囲が広い利用者認証ゲートウェイシステムの開発, 情報処理学会論文誌, Vol.42, No.12 pp.2802-2809 (2001).
- 3) 只木進一, 江藤博文, 渡辺健次, 渡辺義明：利用者移動端末に対応した大規模ネットワークの Opengate による構築と運用, 情報処理学会論文誌, Vol.46, No.4, pp.922-929 (2005).
- 4) 大谷誠, 江口勝彦, 渡辺健次：IPv4/IPv6 デュアルスタックネットワークに対応したネットワーク利用者認証システムの開発, 情報処理学会論文誌, Vol. 47, No. 4, pp. 1146 - 1157 (2006).
- 5) 大谷誠, 江藤博文, 渡辺健次, 只木進一, 渡辺義明：“IPv4/IPv6 に対応したネットワーク利用者認証システム Opengate の改良”, 情報処理学会研究報告, 2006-DSM-43 (2006).
- 6) 大谷 誠, 江藤博文, 渡辺健次, 只木進一, 渡辺 義明：“ HTTP Keep-Alive による利用終了検知機能を実装した新しい Opengate の開発”, 情報処理学会研究報告, 2007-DSM-44 (2007).
- 7) Request for Comments: 2616, Hypertext Transfer Protocol - HTTP/1.1 (1999).
- 8) OneStat.com  
[http://www.onestat.com/html/aboutus\\_press\\_box50microsoft-internet-explorer-7-usage.html](http://www.onestat.com/html/aboutus_press_box50microsoft-internet-explorer-7-usage.html)