Vol.2013-GN-86 No.9
Vol.2013-CDS-6 No.9
2013/1/16

ユーザキャッシュを利用したWebアーカイブの構築

若菜 勇気[1,a)]　長谷川 大[2]　佐久田 博司[2]

概要：インターネットにおいて日々変化し続ける Web ページを後世に残すために，各組織が Web アーカイブに取り組んでいる．Web アーカイブでは Web ページを自動的に探索するクローラを用いてアーカイブを行っている．しかし現状の Web アーカイブでは Web ページを収集するクローラでは静的リンクを辿り Web ページを収集しているため，ブラウザやサーバで動的に生成される深層 Web のコンテンツがアーカイブできない問題がある．そこで本稿ではクローラだけではアーカイブが困難であった Web ページのアーカイブを目的とし，ローカルのユーザキャッシュとクローラで収集されたアーカイブを統合した Web アーカイブを提案する．ユーザキャッシュは動的に生成される Web コンテンツ等，多くの深層 Web のコンテンツが保存されている．そのため提案手法ではより収集率の高い Web アーカイブを構築することが可能である．システムの有用性を示すために深層 Web のコンテンツを含む Web ページにおいて，コンテンツの取得数に関して従来のクローラのみの場合のアーカイブとの比較を行った．その結果，本システムでは外部サイトの API で生成された画像ファイルや，サーバで動的に生成されたテキストファイルなどのアーカイブが可能であることを確認した．

キーワード：Web アーカイブ，グループウェア，ローカルプロキシ

Construction of the Web Archive Using User Cash

Abstract: To leave web contents on Internet, which are changing on every day, to posterity, many organizations are working on archiving them. The web archive has been conducted by using web crawlers. The conventional web crawlers, however, only search web pages by following links written on html files and can only collect static web contents. Therefore, the contents so called the Deep Web, which are dynamically generated on web browsers or on servers, are not archived by the crawlers. In this paper, to successfully archive the Deep Web along with the static contents, we propose a novel archiving system that integrates contents retrieved by a web crawler and from user cashes. The user caches store the Deep Web when users accessed them and the contents were dynamically generated. Therefore, by using user cashes the system can create a web archive with higher reproducibility. To evaluate archive performance, we compared our system with a conventional crawler on the number of contents successfully archived from a web page that contains the Deep Web contents. As results, we confirmed that our proposed system could collect the larger number of contents; especially picture files generated by using API of the outside sites and text files generated on server-side.

Keywords: Web Archive, Groupware, Local Proxy

1. はじめに

デジタルメディアの普及とともに，インターネットでは常に膨大な Web ページが，それを利用するユーザ によって，作成・更新・削除されている．日々 変わりゆく Web ページを後世に残すために，各組織が Web アーカイブに取り組んでいる．アメリカの Internet Archive では 1996 年から一貫して，世界中の Web ページの収集を行っている [2]．日本ではその役目を国立国会図書館が担い，WARP という事業として国内の Web アーカイブを行っている [7]．このようなグローバルな Web アーカイブでは，世界，国

[1] 青山学院大学大学院理工学研究科
Graduate School of Science and Engineering, Aoyama Gakuin University
[2] 青山学院大学理工学部
Department of Science and Engineering, Aoyama Gakuin University
[a)] c5611146@aoyama.jp

内と非常に大きな規模で Web アーカイブを展開している．Web アーカイブでは， Web の膨大な情報を収集するために，Web ページを探索する クローラを用いて Web ページの収集を行っている．しかしクローラだけでは深層 Web に存在するような Web ページのアーカイブが困難である．深層 Web（Deep Web）とは，JavaScript 等で動作する動的コンテンツや，どの Web ページからもリンクの貼られていない Web ページなどを指す [4]．深層 Web の存在により Web 全体のアーカイブは難しく，Web アーカイブを構築する課題となっている．加えて，国家単位で行われている Web アーカイブでは，このようなコンテンツの存在や更新時間などの関係で Web ページを網羅的に保存することは困難である．

そこで本研究では，インターネットを利用するユーザのインターネットへのアクセスデータに着目する．本稿ではこれらのデータをユーザキャッシュと定義する．ユーザキャッシュには，クローラで収集することが難しい動的コンテンツなどの Web ページが含まれている．提案手法ではユーザキャッシュとクローラで収集した Web 情報を組み合わせることで，より詳細な Web アーカイブの構築が可能であると考えられる．

本稿では第二章で深層 Web の性質について述べる．第三章において現状の Web アーカイブの問題点について述べる，第四章で具体的な提案システムについて説明する，提案したシステムについて第五章で実験と結果を述べ，第六章で提案手法の考察を述べる．

2. 深層 Web

深層 Web は，検索エンジン等で利用されているクローラ等では辿り着けない領域に存在する Web のリソースを指す [1]．Web の深い領域に存在する Web のリソースは，Web ページ同士のリンク関係を辿り，Web ページを自動的に収集するクローラには収集が難しい．深層 Web として定義される Web ページは以下の 4 つに分けられる．

(1) 静的なリンクが存在しない Web ページ
(2) データベースから動的に生成される Web ページ
(3) Flash 等で構成された Web ページ
(4) パスワード認証が必要な Web ページ

(1) は Dynamic html などで生成される Web ページなどを指し，Ajax などの技術が昨今発展していく中で増加傾向にある．(2) は，ショッピングサイトなどのデータベースに問い合わせを行った結果，動的にレスポンスが生成される Web ページを指す．現状深層 Web に存在する Web ページ群の中でも数としては大きな母数を持っている [5][10][3]．(3) は動画共有サイトなどで生み出される Adobe Flash などの動画コンテンツなどが該当する．動画コンテンツはひとつのファイル容量が大きく，深層 Web に存在する Web ページの容量として最大規模である．(4) は SNS や EC サイトなどにおけるパスワード認証 (HTTPS や BASIC 認証等) を必要とする Web ページを指す．この Web ページは主に個人情報を取り扱う Web ページなどに適用されることが多いこともあり，取り扱いが難しい Web ページでもある．

2.1 Web アーカイブにおける深層 Web

Web アーカイビングは，基本的に Web ページを自動的に収集するロボットであるクローラを用いてアーカイビングされている．そのため，検索エンジンと同じように深層 Web の Web ページ群をインデックシング，収集する必要性がある．さらに Web アーカイブのクローラには，提供する Web アーカイビングシステムにおいて再現性のある形で Web ページを閲覧できるようにする収集，アーカイブコンテンツ用に保存する技術が必要である．例えばアメリカの Internet Archive が提供している Wayback Machine[2] では，アーカイブされた Web ページが時系列順に保存され，それらの Web ページは再現性のある形で Wayback Machine を通して閲覧することが可能である．ここで述べる再現性とは，Web ページがある特定の時間にブラウザで閲覧したものと，保存された Web アーカイブとを比較し，変化のない形で保存することを指す．

3. 既往の Web アーカイブ

Internet Archive のような大きな Web アーカイブでは，非常に幅広い範囲の Web ページを収集することができている．しかしながら，クローラがたどり着く領域がアーカイブの収集限界であり，更新時間によっては Web のアーカイブを行えていないものも多く存在する．そのため，閲覧したかった過去の Web ページが残っていないということがある．そこで Web アーカイブには国家単位より比較的小規模で行われるものが多いローカルな Web アーカイブも多く提案されている [11]．ローカル Web アーカイブの一つとして，HTTP アクセスの際にブラウザが保存する Web ページのキャッシュを保存する Web アーカイブシステムがある．王らはユーザの各 PC のキャッシュを P2P により分散的に管理することで，仮想的に一つの Web アーカイブを作りだす Web アーカイブを構築した [6]．このような P2P を利用する Web アーカイブは中央サーバを介さないためコスト削減や Web アクセスの高速化を行うことができる [8]．しかしこのようなシステムはあくまでキャッシュの共有による履歴保存システムの側面が強く，Web ページの網羅的なアーカイブには適さない．そのため，ユーザキャッシュを収集した Web アーカイブは履歴保存サービスのように，断片的な形でしかデータが残らない．クローラなどによる Web ページ収集の軸がなければ，網羅的な Web アーカイブにはキャッシュ情報だけでは不十分である [9]．前述したようなキャッシュ共有システムなどのロー

カルな Web アーカイブでは，グループ内において柔軟な設計を行うことができるメリットがあり，国家単位で行うには難しい Web アーカイブの構築をすることができる [9].

4. 提案手法

本手法ではクローラによる収集した Web アーカイブにユーザキャッシュを統合する．ユーザキャッシュには多くの深層 Web のコンテンツが存在する．そのためクローラにより収集されたアーカイブを軸にユーザキャッシュを加えることで，より詳細な Web アーカイブを構築することができる．

4.1 手法の概要

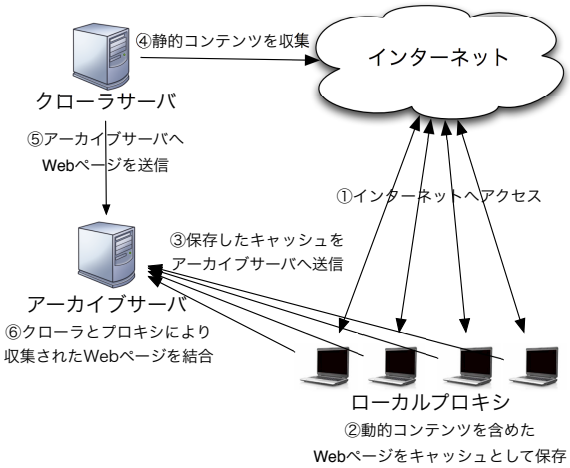

図 1 システムの概略図

ここでは本システムの具体的な実装について説明する．開発した Web アーカイブシステムの概略図を図 1 に示す．本 Web アーカイブシステムはそれぞれ以下のような構成で構築される．

(1) クローラサーバ：ロボットによる Web ページの自動収集
(2) ローカルプロキシサーバ：ユーザキャッシュの自動収集
(3) アーカイブサーバ：上記のデータ統合と閲覧，及びその他認証処理等

各サーバの詳細な動作に関してこれ以降説明を行う．

4.2 クローラサーバの動作

クローラサーバではロボットにより，Web 上に存在する Web ページを自動的に収集，アーカイブデータ用に保存する処理を行う．本システムでは Web ページを探索，収集するクローラとしてオープンソースの Heritrix[*1] を用いる．Heritrix とは，Internet Archive が開発している Web アーカイブ用のクローラである．Internet Archive が運営する Wayback Machine では，実際に Heitrix を用いて Web ページのアーカイブを行っている．このクローラは，通常の検索エンジンにおけるクローラとは異なり，Web のアーカイビングに特化した作りとなっている．クローラは収集した Web ページを，Heritrix 特有のファイル形式である WARC 形式でアーカイブコンテンツ用に保存する．

[*1] https://webarchive.jira.com/wiki/display/Heritrix/Heritrix

4.2.1 WARC ファイル

```
WARC/0.17
WARC-Type: warcinfo
WARC-Date: 2008-04-30T20:48:25Z
WARC-Filename: IAH-20080430204825-00000-blackbook.warc.gz
WARC-Record-ID: <urn:uuid:35f02b38-eb19-4f0d-86e4-bfe95815069c>
Content-Type: application/warc-fields
Content-Length: 483

software: Heritrix/@VERSION@ http://crawler.archive.org
ip: 192.168.1.13
hostname: blackbook
format: WARC File Format 0.17
conformsTo: http://crawler.archive.org/warc/0.17/WARC0.17ISO.doc
operator: Admin
isPartOf: archive.org-shallow
created: 2008-04-30T20:48:24Z
description: archive.org shallow
robots: classic
http-header-user-agent: Mozilla/5.0
http-header-from: archive-crawler-agent@lists.sourceforge.net
```

図 2 WARC ファイルのフォーマット:header 部

WARC ファイルは Internet Archive が Web アーカイブの標準規格として採用しているフォーマットである．WARC ファイルのフォーマットはファイルの先頭に書き込まれる header 部 (図 2) と，複数のアーカイブしたファイルの body 部 (図 3) に分けられる．

header 部は，WARC ファイルが初めて生成される際に WARC ファイルのトップに書き込まれる．ファイル名ごとにハッシュ値と更新時間がヘッダー部に存在する．これにより，WARC ファイルの一意性を保証することが可能である．

```
WARC/0.17
WARC-Type: response
WARC-Target-URI: http://www.archive.org/robots.txt
WARC-Date: 2008-04-30T20:48:25Z
WARC-Payload-Digest: sha1:SUCGMUVXDKVB5CS2NL4R4JABNX7K466U
WARC-IP-Address: 207.241.229.39
WARC-Record-ID: <urn:uuid:e7c9eff8-f5bc-4aeb-b3d2-9d3df99afb30>
Content-Type: application/http; msgtype=response
Content-Length: 782

HTTP/1.1 200 OK
Date: Wed, 30 Apr 2008 20:48:24 GMT
Server: Apache/2.0.54
Last-Modified: Sat, 02 Feb 2008 19:40:44 GMT
ETag: "47c3-1d3-11134700"
Accept-Ranges: bytes
Content-Length: 467
Connection: close
Content-Type: text/plain; charset=UTF-8
```

図 3 WARC ファイルのフォーマット:body 部

body 部では，各ファイルごとにレスポンスヘッダーと

レスポンスボディが WARC ファイルに書き込まれる．なお，HTTP アクセスにおけるリクエストとレスポンスそれぞれに対して，WARC ファイルへの書き込みが行われる．body 部に関しても header 部と同様に，一意なファイル情報が書き込まれたことを示すためにハッシュ値と更新時間が与えられる．

header 部と body 部のフォーマットは，アーカイブ統合処理をする際に重要となる．これはクローラアーカイブと，ローカルアーカイブがともに WARC 形式のフォーマットでファイルの形で統合される必要がある．統合アーカイブは WARC による共通のフォーマットにより，Web ブラウザで時系列順に閲覧することが可能となる．

4.2.2 クローリング方法

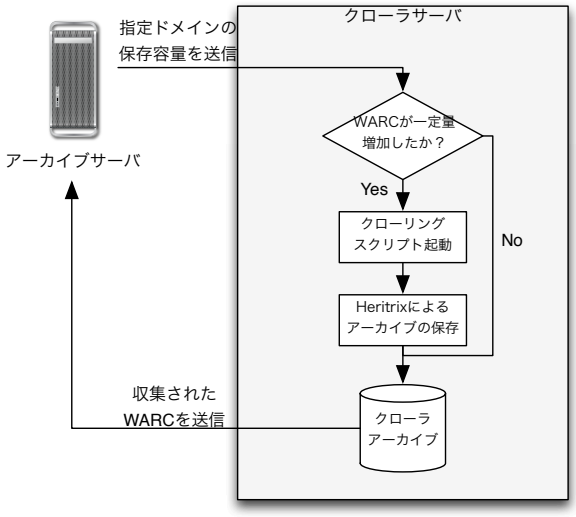

図 4 クローリングのフローチャート

クローラは一定量の Web ページが収集されると再び指定されたドメインに対して Web ページのクローリングを開始する．本システムのクローリングは，ユーザが設定した一定量の Web アーカイブの情報が変更にあった場合にのみ再クローリングを行う．

クローリングのライフサイクルを図 4 に示す．クローラサーバは一定時間ごとにアーカイブサーバより，指定ドメインの更新コンテンツ量の観測データが受信される．クローラサーバは観測データを受信すると，ユーザが指定した一定量の Web ページの書き換えが行われていた場合に，最初に設定した条件と同じ条件でクローラによる Web ページの収集を行う．このクローリングにより保存された Web アーカイブのデータ群は，クローラサーバに保存されたと同時にアーカイブサーバへ送信される．

4.3 ローカルプロキシの動作

ローカルプロキシの動作概略図を図 5 と各スレッドにおけるオブジェクトのシーケンスを図 6 に示す．ローカルプロキシは，クライアントマシンが HTTP アクセスをするごとにそれをトラップする．トラップされたリクエスト，レスポンスは，各スレッドにより特定の処理がなされる．各スレッドの役割について Proxy，Storage，Reception と Warc の順に述べる．

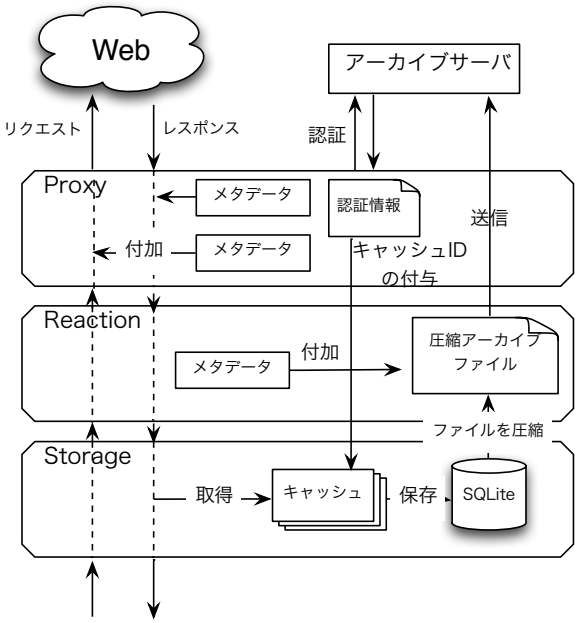

図 5 ローカルプロキシの動作

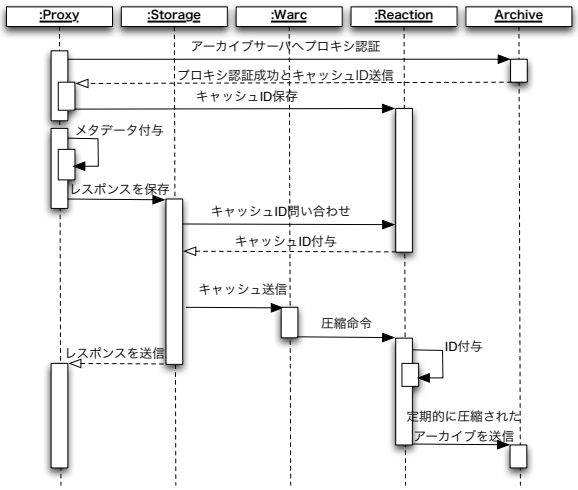

図 6 ローカルプロキシのレスポンスシーケンス図

Proxy スレッドでは，取得したリクエスト，レスポンスのそれぞれに対して，アーカイブデータ用にヘッダーに更新時間やキャッシュ ID を付与する．キャッシュ ID とは，各クライアントマシン上のキャッシュファイルに一意にセットされる ID である．これはアーカイブサーバでの統合処理を行う際に利用される．この ID は各キャッシュの HTTP ボディにセットされている値からハッシュ値と

して算出されるため，同じキャッシュの保存を防ぐ目的でも利用される．ローカルプロキシでは Proxy スレッドにより，起動時に各クライアントマシンごとにハッシュ値によるノード番号が付与される．これにより，クライアントマシンがアーカイブマシンにユーザキャッシュを送信する際に，どのノードがどのキャッシュをアーカイビングしたのか管理することができる．

Storage スレッドは各ユーザキャッシュにメタデータが付加された後，そのデータをデータベースに保存する．データベースには，指定されたドメインごとに Web ページのキャッシュが保存される．

これらの保存されたユーザキャッシュは一定量が保存されると，Reaction スレッドにより自動的にファイルが圧縮・アーカイブ化され，アーカイブサーバ上に送信される．Reaction スレッドと Warc スレッドは任意のタイミングで動作する．Warc は WARC フォーマット用にファイルを変換する処理を行う．なお，アーカイビングしたい Web ページはユーザが指定したドメイン名ごとに行う．もし，リファラーに対して，同じドメイン名を持たない場合はホスト名をさらにチェックする．ユーザが指定したドメイン名をリファラーもしくはホスト名にもつファイルがキャッシュとしてデータベースに保存される．

4.3.1 ユーザキャッシュの保存

保存されるユーザキャッシュは二つの種類に分別される．

(1) ユーザが指定したドメインに一致する Web ページ
(2) それ以外の Web ページ

(1) のは，ローカルプロキシソフトウェアをユーザが起動した際に設定する．ユーザは Web 上に存在するドメイン名（例．www.aoyama.ac.jp）をローカルプロキシソフトウェアに設定することで，ローカルプロキシは指定されたドメイン名をトップレベルドメインとする Web ページを，(2) により生成される通常のキャッシュデータベースとは別の形式で保存する．この場合のキャッシュファイルはレスポンスボディのみならず，レスポンスヘッダーを別ファイルでデータベースに保存する．

4.3.2 ユーザキャッシュの送信

ローカルのデータベースに保存されたキャッシュはユーザが指定したドメイン名ごとに保存されている．これらの Web ページが一定量に達すると，ローカルディレクトリを監視するスレッドがそれを補足する．次に，これらのドメイン名ごとに保存されたファイルは tar.gz 形式で圧縮される．圧縮されたファイルはハッシュ ID を付与された状態で，アーカイブサーバへ送信される．

4.4 アーカイブサーバの動作

4.4.1 アーカイブサーバの概要

アーカイブサーバでは，クローラサーバとローカルプロキシで収集された Web ページの統合処理を定期的に行う．統合処理を行うタイミングは，クローラサーバによって収集された Web ページに対して，一定比率の Web ページが統合された際に行う．アーカイブサーバはクローラサーバにクローラを動作するように指示する．

4.4.2 統合処理

ここではローカルアーカイブとクローラアーカイブとを統合する処理について説明する．ここで述べるローカルアーカイブとは，ローカルプロキシから送信された Web アーカイブファイルである．これはクローラサーバのアーカイブファイルと区別するためにここでは呼称している．

(1) ローカルプロキシで収集されたアーカイブファイルを展開する
(2) ローカルアーカイブ内の各ファイルの URL とボディ部をクローラサーバで収集したファイルと照合する
(3) 照合結果が同じファイルと判定された場合はファイルの変更はしないで (2) へ
(4) 照合結果が異なる場合にはファイルの情報を上書きする
(5) 作業中の WARC ファイルが一定量を超えた場合新たに WARC ファイルを作成する
(6) (1)〜(5) をファイルがなくなるまで繰り返す

基本的に上記のようなファイルチェックを繰り返しを行う．統合処理の途中でファイルが到着した場合，更新時間によってはそのファイルも同時に統合処理を行う．動的に生成される Web ページ関しては，URL が異なっていてもファイルの内容は変わらない場合がある．そのため，特別大きい Web ページ以外はボディ部まで全文一致によるファイルの確認を行う．統合後のファイルは後述する Wayback で閲覧できるように任意のディレクトリに保存される．

4.4.3 Wayback によるアーカイブの閲覧

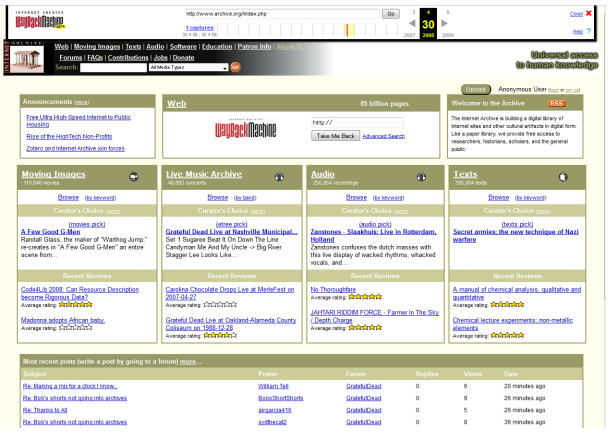

図 7 Wayback による Web アーカイブの閲覧

本システムにより収集された Web ページは前述した WARC 形式で保存されている，WARC 形式にフォーマットされたアーカイブファイルは，インターネット・アーカ

表 1　実験用環境

Web ブラウザ	Firefox 17.0.1
テスト用 Web サイト	外部サイトの API 用いた商品注文サイト
Web サイト開発言語	PHP，JavaScript
Web クローラ	Heritrix 3.1.1
LAN	100BASE-TX

イブがオープンソースで提供している Wayback[*2] というソフトウェアにより，ブラウザ上で閲覧することが可能である．任意のディレクトリに保存された WARC ファイルは，Wayback による検索エンジンに URL を指定することで，時系列順に閲覧できる (図 7)．

しかし，保存された Web ページの中には Wayback では閲覧不可能なデータも存在する．そのため Wayback での閲覧不可の Web ページに関しては別途アーカイブ検索システムを用意した．

5. 実験と結果

本稿で開発した Web アーカイブシステムの評価を行うため，Web コンテンツ収集実験を行った．実験では深層 Web における Web ページがアーカイブされているか，その Web ページの取得数を提案手法とクローラのみ場合と比較する実験である．Web コンテンツ収集実験では前述した二つの手法において，深層 Web の Web コンテンツの取得数について比較を行う．なお，深層 Web の Web コンテンツについては第二章を参照されたい．

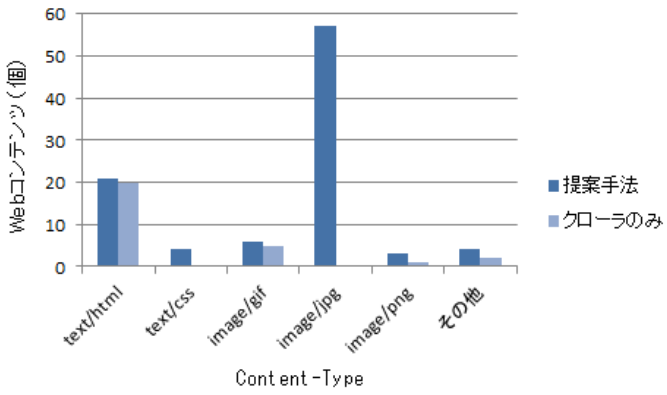

図 8　本システムとクローラのみの場合での Web ページ取得数の比較

実験で使用した Web サイトと，Web ブラウザの環境は表 1 のとおりである．試験用 Web サイトはで PHP と JavaScript により動的に Web ページが生成される．この試験用 Web サイトは一般の Web サービスのように，サーバ上のデータベースに保存された Web コンテンツを，フォームなどのクライアントからのリクエストにより生成されるレスポンスを返す．なお，試験用 Web サイトでは Web ブラウザのキャッシュを無効にした状態で，一度だけ Web ブザウザ上からアクセスした．

本稿で開発したシステムとクローラのみの場合とでの Web ページ取得数の差を図 8 に示す．このグラフでは二つの収集方法において，HTTPHeader の要素である Content-Length により Web ページを分別して集計している．本実験では比較する対象のアーカイブとして Heritrix3.1.1 で収集された Web ページを比較対象とした．

6. 考察

本章では Web 取得比較実験により得られた結果を考察する．加えて，ユーザがローカルプロキシを使用する際に HTTP アクセスにおけるアクセス実験について述べる．これはユーザが通常時の Web アクセスに比べ，ローカルプロキシを使用した際にどの程度アクセス速度に影響が出るか調査する．

6.1 提案手法の深層 Web 取得数について

Web コンテンツ収集実験で収集された Web コンテンツの数の違いについて説明する．どちらの収集方式においても text/html 形式のファイルは双方の環境においても取得できる差はほとんどなかった．ただし，PHP ファイルによりローカル側でレスポンスを受け取ることによって生成される index.html ファイルに関しては提案手法でのみ取得できた．しかしながら，同様に取得できるはずの text/css に関してはクローラ単体では取得できなかった．これは text/css 形式のファイルが，サーバ上の PHP のフレームワークによって動的に生成されたことが起因している．各 image 形式のファイルに関しては顕著な結果となった．これは外部サイトの API により生成される Web コンテンツである．外部サイトで生成される Web コンテンツは非同期アクセスなどにより Web ページが読み込まれた後，改めて実行される処理が行われる場合がある．Web ページが遅れて取得されることが要因になり，クローラ単体では取得できなかったと言える．そのため，Web ページの大部分を構成するこれらの画像ファイルはユーザキャッシュなくしては取得できなかった．その他のファイルでは，JavaScript などの Ajax を利用した Web ページが取得できた．ここで取得できた Web コンテンツも前述した JavaScript のライブラリにより生成されるファイルであった．ただし，クローラのみの環境では JavaScript 関連のファイルは一つも取得できず，Web ページの favicon のみを取得するにしか至っていない．ユーザキャッシュによる Web ページの収集は深層 Web のコンテンツの収集に大きく寄与している．その結果本手法ではユーザキャッシュとクローラの組み合わせにより Web アーカイブの密度を高めることができた．

[*2] http://archive-access.sourceforge.net/projects/wayback/

6.2 ローカルプロキシにおける HTTP アクセス遅延

プロキシ使用時と未使用時における HTTP アクセスの比較を図 9 に示す．実験で使用した Web サイトと，Web ブラウザの環境は第五章の表 1 と同様である．レスポンタイムの比較実験では，レスポンスとして 80 個の Web コンテンツが処理されるまでの時間を計測した．なお，本実験では Web ブラウザによるキャッシュ機能は無効にしている．各環境におけるアクセス時間はプロキシ使用時が 2.250 秒であり，通常時は 1.357 秒であった．このアクセス時間の差は．プロキシによるキャッシュ ID の問い合わせとユーザキャッシュの保存による遅延の差である．提案手法では通常時と比べ，取得されるファイルによらず，各レスポンス処理において概ね 1.5，6 倍ほど時間がかかっている．これによるアクセス遅延は，常にユーザキャッシュを保存するようにローカルプロキシが動作すると，ユーザにとって若干のストレスになり得るかもしれない．しかし提案手法ではすべてのアクセスをキャッシュするわけでなく，普段利用する際は Web ブラウザのキャッシュ機能により緩和されることもあり，使用にそこまで問題がある数字ではないと言える．

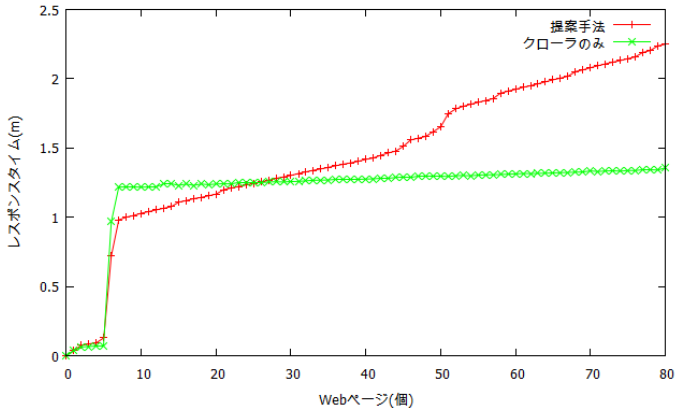

図 9 プロキシ使用時と未使用時の HTTP レスポンスタイムの比較

7. おわりに

本稿では、ユーザキャッシュとクローラアーカイブを統合する Web アーカイブシステムを提案した．評価実験として，深層 Web の取得実験とローカルプロキシの HTTP アクセス実験を行い，本システムが深層 Web のアーカイブに有効性があることを示した．しかし，ユーザキャッシュによる動的に生成される Web コンテンツはユーザのアクセスタイミングによって，アーカイブに偏りが生じてしまう可能性がある．そのためユーザキャッシュの収集に応じてリクエストを動的に生成することによってアーカイブの偏りを減らすことで，より収集率の高い Web アーカイブを構築することが今後の課題である．

参考文献

[1] A. Ntoulas, P. Zerfos and J. Cho: “ Downloading Textual Hidden Web Content through Keyword Queries ”, In Proc. of JCDL2005, pp.100-109, Denver, USA, 2005.
[2] Internet Archive, http://archive.org/index.php
[3] M. lvarez, J. Raposo, A. Pan, F. Cacheda, F. Bellas and B. Carneiro: “ Crawling the Content Hidden Behind Web Forms ”, In Proc. of Int. Conf. on Computational Science and Its Applications, Vol.4706, pp.322-333, Berlin, Heidelberg, 2007.
[4] M.K. Bergman: “ The Deep Web: Surfacing Hidden Value. ”, J. of Electronic Publishing, Vol.7, No.1, 2001.
[5] P. Wu, J.R. Wen, H. Liu and W.Y. Ma: “ Query Selection Techniques for Efficient Crawling of Structured Web Sources ”, In Proc. of the 22nd Int. Conf. on Data Engineering, p.47, Atlanta, GA, 2006.
[6] 王亮，圭博川原，徹浅見. ユーザのキャッシュ情報を活用した分散型ウェブアーカイブシステムの構成. 電子情報通信学会ソサイエティ大会講演論文集，Vol. 2008，No. 2，p. 70，2008.
[7] 国立図書館インターネット資料収集保存事業，http://warp.da.ndl.go.jp/search/
[8] 武晋辻下，俊矢子安，秀輝島田，隆浩小板，健哉佐藤. p2p web キャッシュ共有システムの高効率化の提案. 全国大会講演論文集，Vol. 72，No. 3，pp. 3—255，2010.
[9] 柊和佑，阪口哲男，杉本重雄. 分割・統合可能な組織内 web アーカイブシステムの構成方法. 情報知識学会誌，Vol. 18，No. 1，pp. 47—57，2008.
[10] 舟橋卓也，上田高徳，平手勇宇，山名早人. 商用検索エンジンの検索結果では取得できないランキング下位部分の収集・解析. 日本データベース学会論文誌，Vol. 7，No. 1，p.37-42，2008.
[11] 吉川晃生，阪口哲男. 閲覧履歴を用いた個人用 web アーカイブシステム. 情報科学技術フォーラム一般講演論文集，Vol. 3，No. 2，pp. 109—110，2004.