Webアーカイブにおける URN とヒステリシス署名を用いた原本性保証機能の提案

2001MT012 藤本 晃史　　2001MT058 近藤 慎一
指導教員　河野浩之

1　はじめに

現在インターネットの普及は目まぐるしく，その用途は多岐に渡っている．しかし，Web ページを検索しているとよくページが移動していたり削除されているなどで目的のページを見ることができないことが多々ある．従来の紙媒体の情報資源と比較して内容や存在の空間的，時間的な不安定性があるためである．
情報の内容と存在が空間的にも時間的にも不安定であるというインターネット上の問題は紙媒体における国立図書館のようなラスト・リゾートがないことに起因している．
この事態を防ぐための手立てとして，海外では Web 情報を収集し文化遺産として将来世代のために保存していこうという「Web アーカイビング」[1] という取り組みを行う組織が複数存在し，実際に実験等が行われている．また，日本においても国立国会図書館を中心に Web アーカイビングが注目され，実際に小規模ではあるが実験的に運用されている．そうすることにより，インターネット上の情報に紙媒体で実現されてきた，文化や学問の安定的な発展に必要な，先行業績の参照可能性が保証されると考えられている．
しかし，Web アーカイブの主な問題として，法制度や収集方法，蔵書管理等がある．本研究では蔵書管理の過程で課題となる原本性保証 [1] について議論する．

2　原本性保証

原本性は完全性，機密性，見読性の 3 点を充足させることで保証される．完全性，機密性に関しての対策としては，認証技術，暗号技術を有効に活用することが重要である．代表的なものとして電子署名 [2] がある．見読性の充足で重要なことは，保存された Web ページすぐに検索しブラウザに表示することができることである．そのためには各 Web ページに，検索するのに有効な識別子を付与することで問題が解決することを検討する．

3　アーカイブに関する先行研究

3.1　MD5 ハッシュ関数

電子署名の代表的なものとして，PGP[2] の電子署名がある．PGP 電子署名では MD5[3] というハッシュ関数を用い，その関数で計算された値を非共有鍵暗号方式の秘密鍵で暗号化することで電子署名を生成する．MD5 は任意の長さのテキストブロックから 128 ビットの数を生成し，通常 32 桁の 16 進数文字列として表現される．ハッシュ関数は大きなファイルやメッセージ内のわずかな違いを検出する場合に非常に強力なツールとなる．検出の方法は MD5 コードを計算して，あらかじめ計算しておいたコードと比較すれば良い．一致すればそのファイルは変更されていない．理論上は異なる 2 つのファイルが同じ MD5 コードを持つ可能性はあるが，その確率は $\frac{1}{2^{128}}$ で十分に小さい．実際に MD5 の計算を行った際の例を下に示す．

- MD5(本日は晴天なり．)
  =c18a4d0221083205616eff9d2875a133
- MD5(本日は晴天なり?)
  =eaa8272eb0d2a02b06c3f97ee0c157d7
- MD5(本日は晴天なり，)
  =4153a7fd8a94096cd517ebdadf8bcf01
- MD5(本日は晴天なり．)
  =c18a4d0221083205616eff9d2875a133

3.2　ヒステリシス署名

PGP の電子署名の問題点としては，改ざんされて偽造署名をつくられてしまうと検証することが出来なくなってしまうことである．対応策として再署名という手法があるが，一つ一つ署名を書き換えなければならないので，膨大なコストと時間がかかってしまう．そのため PGP 電子署名は長期の保証には向いていない．そこで電子署名の連鎖構造を作って前後の署名に繋がりを持たせるヒステリシス署名 [4] を紹介する．PGP は Web ページ同士で原本性を保証するのに対し，ヒステリシス署名は過去の自身の Web ページを基盤にして原本性を保証するという，時系列的な保証の仕方である．ヒステリシス署名の特徴として，以下の 4 点が挙げられる．

1. 秘密鍵漏洩時の信頼性確保
2. 署名履歴による検証可能
3. 署名履歴の改ざん困難性
4. 時間的順序性の保持

図 1 にヒステリシス署名の流れを図示する．
署名の連鎖構造が出来たら，署名の信頼性をより強固にするために，定期的に最新の署名記録を CD-ROM に保存したり，新聞や官報等の刊行物に公開してトラストアンカー [4] を形成する．

3.3　ハッシュツリー

Web アーカイブでページをうまく整理するために，ページをサイト単位でまとめる必要がある．ハッシュ値を集約して一つのハッシュ値を生成する手法として，セキュアシールの技術で用いているハッシュツリー [5] について説明する．
ハッシュツリーは，Web ページから生成されるハッシュ

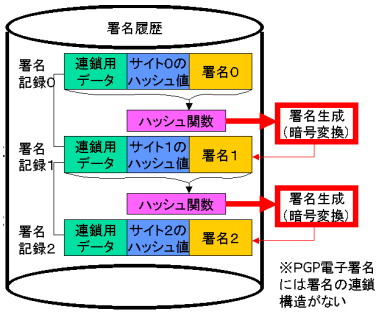

図 1　ヒステリシス署名のモデル

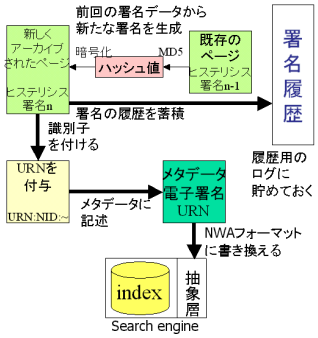

図 3　拡張したエクスポータ概念図

値を集約し，二分木の構造を作りあげている．ハッシュ値が持つ一方向性・不可逆性という特性と，どのようなデータからもハッシュ値が作り出せるという利点を生かし，ページから集まってくるハッシュ値から「ハッシュ値同士を組み合わせたハッシュ値」を次々に作り，最終的にルートハッシュ値という一つのハッシュ値を作り出す．ハッシュツリーのモデルを図 2 に示す．

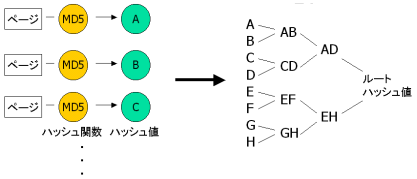

図 2　ハッシュツリーのモデル

3.4　URN

アーカイブした Web ページを容易かつ効果的にアクセスするための手法として，URN(Uniform Resource Name) を用いる．URN を URL(Uniform Resource Locator) と比較すると，URL は Web ページの置いてある場所を指定する方式なので，一意に特定することは難しく，将来的にそのページがなくなったり，移動したときに過去蓄積してきた Web ページと対応させることが困難だが，URN の場合，情報資源の内容的なまとまりを識別するための一意の名前であるから，そのページ自体を指すための識別子と言える．

4　原本性保証機能の提案

4.1　NWA

NWA[6] という Web アーカイブのエクスポータという機能を例にして原本性保証機能を提案する．エクスポータの説明を以下に記す．

エクスポータがメタデータを得た後，そのオブジェクトが html 形式であったら，オブジェクトを得た後データを取り出す．そして，取り出したデータに NWA ドキュメントに適したメタデータを付けて出力する．html 形式でなかった場合 (gif 等) そのデータのメタデータはアーカイブ ID，元あった URL，そのデータ自身のフォーマット，タイムスタンプがメタデータになる．

4.1.1　NWA の問題点

NWA では，エクスポータで Web ページにハッシュ関数 MD5 によりハッシュ値を生成しているだけなので，機密性，完全性が充足していない．

また，ページの識別子はメタデータのハッシュ値をそのまま使用しているため，見たいページを見ようとする時に，そのページのハッシュ値を打ち込まなければならず，直ちに閲覧出来るとは言い難い．キーワードで検出することは出来るが，図書館の蔵書のように内容のわかる名前が無いので見読性の観点から見ても不十分である．

そこで，この問題を解決するために，エクスポータの機能を拡張することを提案する．まずヒステリシス署名を用いて完全性と機密性の充足を図り，その後署名を付けたページに URN を付与し，それをメタデータに記述する．拡張したエクスポータの概念図を図 3 に示す．

4.2　ハッシュツリーを利用したヒステリシス署名

取得した Web ページの完全性・機密性を確保する上で，以下の二つの点が課題になる．

1. サイト単位の保証
2. 時系列での保証 (長期保存)

時系列での保証をするためにヒステリシス署名を，サイト内全てのページの保証をするためにハッシュツリーを用いる．まずハッシュツリーでルートハッシュ値を生成する．構成する全てのページから生成されたハッシュ値に依存しているため，どれか一つのページのハッシュ値が改ざんされたら，同一のルートハッシュ値を生成することは不可能となる．

次に時系列的な原本性の保証についてヒステリシス署名を使う．ルートハッシュ値が初めて取得した Web ページの場合，秘密鍵を用いて署名を作る．この署名は署名履歴に保存しておく．次にそのサイトが更新され，再収集が行われたページのとき，ヒステリシス署名は以前に作った署名を使う．これを再収集したページに添付する．そして，収集した場合の署名の作成を行う．今収集してきたサイトのルートハッシュ値を計算する．そして

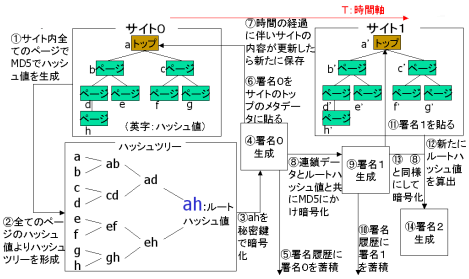

図 4　署名生成の流れ

そのルートハッシュ値にヒステリシス署名を張り付け，ハッシュ値を計算し，暗号化して新たなヒステリシス署名を作成する．そして作ったヒステリシス署名を署名履歴に保存する．図 4 に署名生成の流れを示す．

4.3　アーカイブに用いる URN の提案

図書資料の分類分けと Web ページの識別子の対応を表 1 に記す．

書籍ではなく Web サイトを識別するための URN を付けるため，書籍の識別子と Web ページが対応する URN を考察する．

4.3.1　Namespace ID

収集したアーカイブのデータベース名は収集を行っているアーカイブ組織の名前である．各々のアーカイブ機関独自の URN が必要になることが考えられるためアーカイブ機関を識別することは必要になる．これを NID として採用することを提案する．

4.3.2　対象範囲

収集したサイトの地理的な場所での区別をする．ここで，日本の歴史に関する記述のされているサイトがアメリカ等のサーバ上に置いてあるという場合，置いてある場所という括りではアメリカの分類になってしまう．この場合はアーカイブの目的と使用上の都合などを考えて言語が日本語でなくとも日本の Web ページという括りの中に入れることにする．その判断はそのサイトに含まれる単語などから判断する．コードに関しては NBN と同様に ISO3166 国コードの接頭辞を用いることにする．例えば，日本は jp，アメリカは us である．

4.3.3　公開者

サイトを管理している団体の識別については，大学のページ (ac) や企業のサイト (co) は，その団体がページの管理を行っている．個人が管理する場合については gTLD の name を使用する．また jp は日本独自のものなので各国の場合はその国に対応している属性ドメインを使用する．また特に規定のない国の場合は gTLD で対応する．

4.3.4　主題

主題に関しては書籍の分野の分類に用いられている NDC(Nippon Decimal Classification)[7] を採用する．分類方法はクラスタリングの技術を用いる．NDC を採用する利点は同じ分野を一箇所にまとめておくことができる点である．NDC の例として，南山大学のページなら大学・高等・専門教育にあたる 377 である．NDC は日本独自のものであるため，各国のアーカイブで用いる場合には国ごとの分類コードを用いる．例えば，LCC(Library of Congress Classification，米国議会図書館分類表) や UDC(Universal Decimal Classification，国際十進分類法) 等がある．

4.3.5　タイトル

ページのタイトルは収集してきたサイトのタイトルを付ける．ここで，一口にサイトと言ってもその括りは大変難しく明確に区別することができない．そこで本研究でのサイトの概念は，同一ドメイン配下にある全ての情報かつ起点 URL 配下の全ての情報とする．

またタイトルの記述がないページは，各分野に識別する際に抽出したキーワードを要約として使用する．また画像や音楽等の非テキスト形式のページの場合はそのファイル名をタイトルとして使用する．

4.3.6　日付

日付は出版物における巻数，号数や版数に相当するもので，最初に収集した日付または再収集を行った日付を指す．URN には年月日までの表時にするが，メタデータには収集した年月日に加えて収集作業を開始した時間を秒単位で保存しておき，もし同じ日時に収集するようなことがあったとしても区別できるようにする．日付の記述形式は W3C-DTF に基づく．W3C-DTF は ISO-8601 のサブセットとして，日時の表記方法を限定西暦で 4 桁，月 2 桁，日 2 桁をハイフンで結んで表現する．Y：西暦の数字，M：月の数字，D：日の数字を表す (YYYY-MM-DD)．例：2004-12-01

4.3.7　例

以上の提案により，URN の記述は以下となる．例に挙げるページは南山大学のページに URN を付けた場合である．収集日時は 2003 年 12 月 14 日である．

> URN:アーカイブ名:jp-ac-377-南山大学-2003-12-14

4.4　提案するメタデータ

NWA のメタデータ XML フォーマットを基盤とした提案するメタデータを記述する．

表 1　書籍の識別子と Web ページの識別子の対応

要素	書籍の識別子	URN の識別子
機関名	収録データベース	収集したアーカイブのデータベース名
対象範囲	出版された本の地理的な場所 (国や地域)	収集した Web ページの地理的な場所 (国や地域)
公開者	著者もしくは寄付者	サイトを管理している団体
主題	学術論文や技術論文等	Web サイトの内容的な分類
タイトル	本のタイトル (論文名や雑誌名)	サイトのタイトル
日付	出版日，(巻数，版数)	作成日，公開日，更新日，収集日

提案するメタデータ XML フォーマット

```
<metadata>
  <url> www.nanzan-u.ac.jp </url> */取 得
URL/*
  <time> 20050103030233 </time> */取 得 時
間/*
  <contenttype> */コンテント形式/*
    <type> text/html <type> */記述形式/*
    <charset> shift_jis </charset> */文
字コード/*
  </contenttype>
 <http-header> */ヘッダ/*
 <SignedInfo> */署名情報/*
    <SignetureMethod> Hysterisis
    </SignetureMethod>*/署名方式/*
    <HashMethod> MD5 </HashMethod>
    <HashValue> 4e24501f9084f956e985176
    2b924ff47 </HashValue> */ハッシュ値/*
   (<RootHashValue> 89beff36074c04543da8ee
    a690473794  </RootHashValue>) */ル ー
トハッシュ値/*
    <SignatureValue> iQA/AwUBQjZtjzq1qhl
    q2PEQJ1IQCeIebf0Aykhu30qgYoAQu/PaleS
    xwAoNDx1jdHIjLp95U8z7xZ3WLrq30=miQR
    </SignatureValue>*/ヒステリシス署名/*
  </SignedInfo>
  <SigningTime> Mon, 03 Jan 2005 03:02:33
  </SigningTime>*/署名時間/*

  <UniformResourceName> URN:アーカイブ名:
  jp-ac-377-南山大学-2003-12-14
  </UniformResourceName>*/URN/*

    ヘッダ情報

  </http-header>
</metadata>
```

追加した部分は，署名情報とその下位の要素，署名時間，URN と URN を付けた時間である．

5　おわりに

ハッシュツリーを利用したヒステリシス署名を用いれば長期間の原本性保証が可能になり，またサイト内のページの検出も強力になる．そして署名を付けたサイトに URN を付けることで，見読性も向上させることができる．

しかし，管理者が 2 重に署名の履歴を作ることや，ハッシュツリーを用いたページ単位での改ざんの検出には依然，鍵の証明に有効期間が残ってしまう問題がある．また URN の構想についても実際に URN の管理を行うにあたって「どのような機関が，どのように管理し，どのように保証するのか」をより深く議論する必要がある．

謝辞

本研究を進めるにあたり，河野浩之教授による御指導はもちろん，研究室の仲間にも支えられ，多くのアドバイスを頂きました．関係諸氏に深く感謝します．

参考文献

[1] 広瀬信己:国立図書館におけるウェブ・アーカイビングの実践と課題，社団法人情報処理学会 研究報告 (2003).

[2] About the NWA Toolset http://nwatoolset.sourceforge.net/index.php?doc=aboutNwaToolset(accessed 2004.8.25).

[3] Simson Garfinkel，山本和彦監訳:PGP 暗号メールと電子署名， オライリージャパン (1996)

[4] R.Rivest:The MD5 Message-Digest Algorithm，RFC1321(1992).

[5] 本多 義明:認証技術の新応用， (株) 日立製作所，http://www.japanpkiform.jp/sympo_domestic/material/5_hitachi_honda.pdf(accessed 2004.12.1).

[6] NTT データ経営研究所 編著:電子文書証明 e ドキュメントの原本性確保，NTT 出版 (2001.8.7).

[7] 日本図書館協会: http://www.soc.nii.ac.jp/jla/ (accessed 2004.12.20).