階層型WIX(Web Index)とエントリ重複に関する一考察

森　　良介†　　遠山　元道†

† 慶應義塾大学理工学部情報工学科　〒 223–8522 神奈川県横浜市港北区日吉 3–14–1
E-mail: †{morisi,toyama}@db.ics.keio.ac.jp

あらまし　Web Index（以下 WIX とする）ファイルとは，ユーザ主体で Web 情報の結合を実現するために用いる XML 形式のファイルであり，キーワードとそれに対応する URL からなるエントリを持つ．著者らは従来 WIX ファイルを各々独立したものとして扱っていたが，本研究では WIX ファイル階層構造を構成することを提案する．従来手法による結合結果との比較検討を行い，さらに，WIX ファイルのエントリ重複に対して RDB における JOIN との類比から，種々の結合演算の実行，評価，検討を行った．

キーワード　WIX，Web，ハイパーリンク，XML

Consideration concerning hierarchy WIX(Web Index) and entry repetition

Ryosuke MORI† and Motomichi TOYAMA†

† Department of Information and Computer Science, Faculty of Science and Technology, Keio University
Hiyoshi 3–14–1，Kouhoku-ku，Yokohama-shi，Kanagawa，223–8522 Japan
E-mail: †{morisi,toyama}@db.ics.keio.ac.jp

Abstract　A Web Index（assume it WIX as follows）file is the file of the XML form to use to realize combination of the Web information with the user subject and has an entry consisting of keyword and the URL corresponding to it. We treated a WIX file as an independent thing each conventionally, but we suggest constitute WIX file hierarchical structure in this study. We performed comparison with the combination result by the technique conventionally and examined with practice and the evaluation of various kinds of join operations from analogical inference with JOIN in RDB for the entry repetition of the WIX file.

Key words　WIX，Web，hyperlink，XML

1. はじめに

Web における利用者主導による情報資源の結合を実現するために，我々は Wen Index（以下 WIX とする）と呼ぶ情報資源表現形式の提案，開発を行っている．WIX は見出し語とそれに対応する文書のペア（エントリ）の集合であり，それを閲覧中の Web 文書に結合（アタッチ）すると，閲覧中の文書の見出し語部分が自動的に対応する URL へのハイパーリンクに変換される．

WIX には利用者と生成者がおり，生成者が予めプールしておいたエントリのリストからなる WIX ファイルを利用者が任意に使う形となる．従来，WIX ファイルは各々独立したものとして扱われていたが，本論文ではそれらの WIX ファイルに対し階層構造を構成することを提案する．さらに，それに伴って生じるエントリ重複という問題に対して種種の結合演算を提案し，これらの演算を実行するシステムを作成，評価を行った．

本稿の構成は以下の通りである．まず，2 章で，WIX の概要について述べる．3 章では WIX の記述形式について説明する．4 章では階層型 WIX，5 章では種々の結合演算についての提案・定義を行う．6 章にて提案手法の検討を行い，最後に 7 章で結論を述べる．

2. WIX

ここでは，本研究により演算を行ったファイル出力を用いる WIX の概要について述べる．WIX は Web における利用者主導による情報資源の結合を実現するための情報資源表現形式である．

2.1 背　　景

近年，検索エンジンの普及によって人々は日常的に Web を利用し情報検索を行うようになった．その上で，ユーザは検索エンジンによって必要な情報資源を検索し結果の Web 文書中から必要な情報を得る．Web 上では関連する情報はハイパーリ

ンクで結合される．リンク元の文書とリンク先の文書間の関連はホームページ作成者の意図した関係のみで提供されている．このため，ユーザが Web 文書中で見つけた単語や名称を元に新たな情報を得たいという要求が生じた場合，ユーザはさらにその単語を検索エンジンなどにかけなければならない．

2.2 ア タッチ

基本的な WIX ファイルはエントリと呼ばれる見出し語とそれに対応する文書からなるペアの集合である．見出し語は関係データモデルのキーに相当する．文書はそのキーが属する行のデータに相当し，主に情報源の URL が格納される．

Web 利用者は閲覧中の文書にある山についての詳細な情報を得たい場合，従来では山の名前を一つずつ検索エンジンで検索せねばならない．これに対し WIX では現在の文書に WIX ファイルを結合することで，閲覧中のページにある山の名前から，それぞれの山の情報へのハイパーリンクが一斉に生成される．これは，ある文書から他の文書へのインデックス付けであると見なすことができ，この結合を“ アタッチ ”と呼ぶことにする．

同じ単語をキーとした WIX でも，例えば英和辞書の他にも英仏辞書や英露辞書まで生成者が提供すれば，利用者は必要に応じて自由に WIX を選択し，自分の閲覧している文書にアタッチすることで状況に応じた情報を簡単に切り替えることができる．

WIX はアタッチを行う文書に独立に存在するものであるため，誰でも“ 辞書 ”を作成・公開することができ，Web ページ作成者の意思に関係なく，ユーザ主体で動的に自由な結合を行うことができ汎用的であると言える．

アタッチには以下の三種類がある．

フルアタッチ

キーワードに合致する文書中の全ての単語に対しリンクを生成する

アンカータグによるアタッチ

Web 文書作成者がアタッチを行いたい単語に対しアンカータグを付与し，指定された単語にのみリンクを生成する

ユーザー指定アタッチ

Web 文書を閲覧するユーザがアタッチを行いたい単語を指定し，指定された単語が WIX のキーワードと合致すればリンクを生成する

3. 記 述 方 式

WIX ファイルは XML 形式であるが，その記述方式には大きく分けて 2 種類存在する．項目が実体として存在する静的（Static）方式と，項目の一部がデータベースに格納されている動的（Dynamic）方式の 2 種類である．

3.1 静的 WIX

Static WIX は単純に一つ以上のキーワードとそれに対応する文書のペアを一つのエントリとして記述する．キーワードを <kw> タグ，ドキュメントの情報を <doc> タグで囲いそれらを一つとして <entry> タグで囲う．各エントリの集合を <body> タグの子要素とする．また，<adoc> タグの type 属性の値を static-separate とする．実際の記述例を図 1 に示す．

```
<?xml version="1.0"
encoding="Shift-JIS" standalone="yes" ?>
<adoc type="static-standard">
  <header />
  <body>
    <entry>
      <kw>dexterity</kw>
      <kw>Dex</kw>
      <doc>http://www.XXX.com/AAA.html</doc>
      <doc>http://www.YYY.co.jp/BBB.html</doc>
    </entry>
    <entry>
      ...
    </entry>
  </body>
</adoc>
```

図 1 Static WIX

3.2 動的 WIX

Static WIX はキーワードと文書のペアを一つのエントリとしていたのに対し，佐藤らが研究を進めている Dynamic WIX は，キーワードに対する文書をを持たず，キーワードの集合のみ，または情報検索のためのコード等を持つ．

Dynamic WIX は静的に保持することが困難な情報，例えば気象情報や株式情報などの時系列データを提供するのに適している．

4. 階層型 WIX

本章では WIX ファイルにおける階層構造について述べる．

4.1 階層型 WIX

従来，WIX ファイルは各々独立して扱われ，利用者は必要に応じて WIX ファイルを選択していた．しかし，大規模な WIX ライブラリの場合，一つの WIX ファイルに大量のエントリを記述することはライブラリの保守管理を困難なものとする．その対策として WIX ファイルを分割して管理する手法が考えられるが，ここではさらに，分割した WIX ファイルにおいて階層構造を構成することにより，分割した WIX ファイル同士の関連付けを行うことを提案する．この階層構造を構成した WIX ファイルの構造を階層型 WIX と呼ぶ．この階層型 WIX によって，WIX ファイルは分割管理を行っているにもかかわらず，分割管理を行っていない場合と同等の情報を，分割を意識させることなく利用することが可能になると考える．

なお，親となる WIX ファイルは実体情報となるエントリは持たず，あくまで参照情報のみを持つものとする．

4.2 記述方式

階層型 WIX の記述は，WIX ファイルのヘッダ内に子となる WIX ファイルのパスを記述することによって行う．子 WIX ファイルのパスを <child> タグで囲い，<child> タグを <header> タグの子要素とする．実際の記述例を図 2 に示す．

```
<?xml version="1.0"
encoding="Shift-JIS" standalone="yes" ?>
<adoc type="static-standard">
  <header>
        <child>./AAA.wix<child>
        <child>./bbb/CCC.wix<child>
  <body />
</adoc>
```

図 2　階層型 WIX

階層構造の構成は実体となる WIX ファイル自体には影響を及ぼさないため，複数の親 WIX ファイルが子として同一の WIX ファイルを指定することも可能である．

5. 結合演算

本章では WIX ファイルにおけるエントリ重複という問題を提示した上で，その対応策である種々の集合演算について述べる．

5.1 エントリ重複

階層型 WIX によって，複数の異なる WIX ファイルを関連付け，階層構造を構成することが可能となった．これにより今後は，ファイル管理という観点以外にも，様々な観点から階層構造を構成することも考えられる．この際，同時に使用することを想定して作成されていない WIX ファイルが同時に使用される可能性がある．この場合はこれらの WIX ファイル同士が disjoint とは限らず，エントリの重複が問題となる．単純に WIX ファイルの <body> タグの子要素を結合しただけでは，同一の見出し語と文書を持つエントリという冗長性を持ってしまうためである．そこで著者らは対応策として種々の結合演算を定義する．

5.2 結合演算

5.2.1 集合演算との類比

集合演算との類比より，以下の UNION ALL，UNION，INTERSECTION，DISJOINT という各結合演算を定義する．

UNION ALL
WIX ファイルのエントリ集合を単純に結合する演算である．エントリ重複に対して特に操作は行わない．

UNION
WIX ファイルのエントリ集合の和集合をとる演算である．エントリの重複部分は除去される．すなわち，エントリ集合を結合し，そこから重複するエントリを削除する演算である．

INTERSECTION
WIX ファイルのエントリ集合の積集合をとる演算である．すなわち，エントリ集合から重複が起こったエントリのみを出力する演算である．

DISJOINT
WIX ファイルのエントリ集合の重複なき部分の集合をとる演算である．すなわち，エントリ集合から重複が起こらなかったエントリのみを出力する演算である．

5.2.2 JOIN 演算との類比

上記の演算では見出し語と文書の組が一致する場合に重複として除去を行っていた．SQL の JOIN 演算においては，結合条件を満たさないデータも表示する OUTER JOIN 演算が存在する．これより，著者らは「キーワードとドキュメント双方の一致」という条件が成立しなくとも，どちらかが一致すれば同一エントリにまとめる UNIFY，STRONG INTERSECT，STRONG DISJOINT という各結合演算を定義する．UNIFY 演算の実行例を図 3 に示す．（注：図の WIX ファイルは body 部分の抜粋である）

これらの結合演算の定義により，新たな見出し語と文書の組み合わせが生成し，既存のエントリ集合から新たなエントリを得ることが可能になると考える．

UNIFY
WIX ファイルのエントリ集合において，見出し語もしくは文書が一致するエントリを一つのエントリにまとめて出力する．なお，重複部については除去を行う．

STRONG INTERSECTION
UNIFY においてまとめられたエントリのみを出力する．

STRONG DISJOINT
UNIFY においてまとめられなかったエントリのみを出力する．

GENERATED
UNIFY によって新たに生成した見出し語と文書の組のエントリのみ出力する．

6. 評　　価

本章では提案手法の評価について述べる．

6.1 階層型 WIX について

WIX システムにおいては一つの文書に対して複数の WIX ファイルをアタッチすることが可能である．ここで，WIX ファイルの分散管理のみを行うことを仮定して一つの文書に対して複数の WIX ファイルをフルアタッチに用いた場合と，提案手法によって分散管理を行っている WIX ファイルに対し階層構造を構成し，結合を行った WIX ファイルをフルアタッチに用

```
...
  <body>
    <entry>
      <kw>abc</kw>
      <doc>http://www.XXX.com/iii.html</doc>
    </entry>
    <entry>
      <kw>def</kw>
      <doc>http://www.YYY.co.jp/jjj.html</doc>
    </entry>
  </body>
...
```

```
...
  <body>
    <entry>
      <kw>abc</kw>
      <doc>http://www.YYY.co.jp/jjj.html</doc>
    </entry>
    <entry>
      <kw>ghi</kw>
      <doc>http://www.ZZZ.ac.jp/kkk.html</doc>
    </entry>
  </body>
...
```

```
...
  <body>
    <entry>
      <kw>abc</kw>
      <kw>def</kw>
      <doc>http://www.XXX.com/iii.html</doc>
      <doc>http://www.YYY.co.jp/jjj.html</doc>
    </entry>
    <entry>
      <kw>ghi</kw>
      <doc>http://www.ZZZ.ac.jp/kkk.html</doc>
    </entry>
  </body>
...
```

図 3　UNIFY

いた場合の処理時間の比較を行った．

6.1.1　実 験 環 境

まず，評価用の WIX ファイルとして下記の web ページから日本野球機構の選手情報が格納されている WIX ファイルを作成した．WIX ファイルの作成にあたっては WEB リンク集再利用化ツール [4] を活用した．

- 日本プロ野球 12 球団公式ホームページ [5]〜[16]
- Wikipedia [17]
- Yahoo!カテゴリ [18]

公式ホームページ，Wikipedia から作成した WIX ファイルは一つの見出し語に対し一つのの文書を持ち，Yahoo!カテゴリから作成した WIX ファイルは一つの見出し語が複数の文書を持つ場合もある．元となった Web ページが同じ場合，それぞれの WIX ファイルは disjoint であるが，元となった Web ページがそれぞれ異なる場合は disjoint とは限らない．見出し語と文書の組み合わせの数は公式ホームページから作成した WIX ファイルが合計 1031 エントリ，Wikipedia から作成した WIX ファイルが合計 1122 エントリであり，Yahoo!カテゴリから作成した WIX ファイルが合計 1065 エントリである．

評価に用いた環境は CPU: Intel Core 2 Duo T7500 2.2GHz，メインメモリ: 2GB，OS: Windows Vista Home Basic である．アタッチにあたっては Firefox のアドオンを用いた．アドオンのアタッチアルゴリズムはエントリの見出し語一つ一つに対して，文書中に一致する語句が存在するか探索するナイーブなものである．前章で構成した階層構造のファイル数が 1,2,4,12 である各階層（図 4）において，フルアタッチを 6 回行い，2 回目から 6 回目まで 5 回分の処理時間の平均値をとった．処理時間は 閲覧者がフルアタッチを実行してからブラウザが結果を出力するまでの時間である．

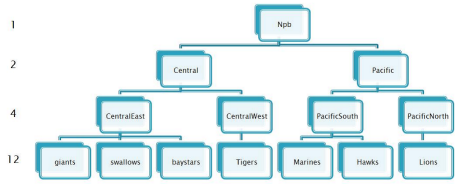

図 4　階層構造（抜粋）

6.1.2　結　　果

上記の環境で行った実験の結果について処理時間を表 1 に，処理時間の標準偏差を表 2 に示す．なお，アタッチによって 15 語 21 リンクが生成した．

表 1　2008 年度個人タイトル受賞者ページに対する処理時間

ファイル数	処理時間（sec）
1	6.2
2	6.3
4	6.4
12	6.9

表 2　2008 年度個人タイトル受賞者ページに対する処理時間の標準偏差

ファイル数	標準偏差（sec）
1	0.0528
2	0.0904
4	0.112
12	0.119

6.1.3　評価・考察

実験結果より，どちらの Web ページについてもファイル数の増加と共に処理時間が増加することがわかる．また，標準偏差を考慮しても，階層構造を構成した場合は構成しない 12 ファイルをアタッチした場合に比べ処理時間が短縮されると言える．このことは先行研究 [3] であるサーバーサイドでの WIX ファイルのアタッチの際にも示されており，一つの WIX ファイルごとに DOM ツリーを生成しエントリのマッチングを行うため，読み込む WIX ファイルの数が増えるほど DOM ツリーを生成するための処理時間が発生することが原因と考えられる．そこで対策として WIX ファイルを結合し，単一ファイルとして処理することが挙げられていたが，本論文の提案手法である階層構造の構成により WIX ファイルの結合が容易に行うことが可能となった．

6.2　結合演算について

前章では様々な結合演算の定義を行った．ここで，結合演算の中から既存の見出し語と文書の組み合わせから新たな見出し語と文書の組み合わせを生み出す UNIFY 演算に注目し，生成したエントリに対して見出し語について文書が対応しているかどうかの適合率について評価を行った．

6.2.1　評 価 手 法

UNIFY 演算によって生成したエントリのみを出力する GENERATED 演算を用いてを調査した．先述の日本野球機構の選手情報が格納された WIX ファイル全てに対して GENERATED 演算を行い，生成したエントリに対して見出し語について文書が対応しているかの検査を行った．

6.2.2　結　　果

上記の適合率調査の結果について，表 3 に示す．

表 3　GENERATE 演算の適合率調査

演算前のエントリ数	生成エントリ数	対応エントリ数	適合率
3218	747	697	0.93

6.2.3　評価，考察

実験結果より，GENERATED 演算によって 93 %という高い適合率で既存のエントリから新たなエントリが生成できたことがわかる．例えば Yahoo!カテゴリを基に作成した WIX ファイルの“ 西岡剛 ”という見出し語が，千葉ロッテマリーンズ・オフィシャルサイトの“ http://www.marines.co.jp/team/player/detail/7.html ”という文書を参照しているため，千葉ロッテマリーンズ・オフィシャルサイトから作成した WIX ファイルの“ 西岡 剛 ”という見出し語と Yahoo!カテゴリから作成した WIX ファイルの“ 西岡剛 ”の文書，さらに Wikipedia から作成した WIX ファイルの“ 西岡剛 ”の文書との結合が起こり，“ 西岡 剛 ”という見出し語から Yahoo!カテゴリの文書や Wikipedia の文書が参照できるようになった．

しかし，WIX ファイルは辞書的役割を果たすものであるため，その適合率は限りなく 100 %に近づけなければならない．ここではさらに，見出し語と文書が正しく対応していないミスマッチとされた 44 のエントリについてさらに評価，考察を行った．

ミスマッチの種類として，リンクが切れているものとリンク先が文書と対応していない情報を示したものの 2 種類が存在する．このうち，リンク先が文書と対応していないものをその原因が Web ページのリンクが間違っているもの，1 つの文書が複数の見出し語に参照されているものの 2 つに分け，全体として 3 つに分類し，それぞれについて原因の数と誤エントリの数の調査を行った．調査結果を表 4 に示す．

表 4　正しく対応していないエントリの分類

	リンク切れ	リンク間違い	複数参照
原因数	2	24	2
誤エントリ数	2	27	22

この結果を踏まえ，それぞれの原因に対して考察を行う．

6.2.4　リンク切れ，Web ページのリンク間違い

リンク切れ，Web ページのリンク間違いは WIX ファイルを作成するにあたって基となった Web ページのリンク切れ，リング間違いが原因として生じる．例えば，東北楽天ゴールデンイーグルス・オフィシャルサイトにおいて，イーグルスの背番号 32 の選手である“ 小坂 誠 ”という見出し語と“ http://www.rakuteneagles.jp/team/player/detail/32.html ”という文書とが対応しているにもかかわらず，Yahoo!カテゴリにおいては背番号 32 を前につけていた“ 沖原佳典 ”という選手の項目でこの文書を参照している場合，“ 小坂 誠 ”という見出し語と“ 沖原佳典 ”の文書が結合されてしまうというミスマッチが生じてしまう．

これらの場合，結合元の WIX ファイルにあるノイズを UNIFY 演算において増幅してしまうという結果になっている．この現象を防ぐため，WIX ファイルを Web ページから作成する際，その Web ページの情報が最新のものであるかどうか確め，作成した後も WIX ファイルの辞書的情報は常に最新のものにしておく必要がある．

6.2.5　複 数 参 照

複数参照による文書と見出し語のミスマッチ例としては，Yahoo!カテゴリにおいて“ 和田毅 ”という見出し語と“ 新垣渚 ”という見出し語が共に“ http://www.ne.jp/asahi/regulus/dream/ ”という文書を参照していた場合，これを媒介として“ 和田毅 ”，“ 和田 毅 ”，“ 新垣渚 ”，“ 新垣 渚 ”全ての情報が結合されてしまいミスマッチが生じてしまうといったような例があげられる．

この問題は，複数参照という原因一つに対して，大量のミスマッチが生じる危険がある上，基となる Web ページや WIX

ファイルには問題がないにもかかわらず生じてしまう．この問題を防ぐために，UNIFY 演算の実行の際は演算に用いる対象の WIX ファイルが持つ文書を選手の個別情報等，他の見出し語から独立したものに限定する必要がある．

6.2.6 その他の問題

本評価実験では生じなかったが，見出し語についても複数参照と同様の現象が生じる可能性も存在する．関係データモデルの主キーに相当する見出し語が，結合演算によって主キーとして働かなくなってしまう場合である．

本評価実験においては見出し語は氏名を用いていたため，結合をした際に主キーの役割を果たさなくなる場合はなかった．しかしそれが氏だけになった場合，例えば“ 松井 ”という見出し語を想定した時，UNIFY 演算を行った際“ 松井秀喜 ”を意味する文書と“ 松井稼頭央 ”を意味する文書とが結合されてしまうという可能性がある．この場合，複数参照の場合と同様，基となる Web ページや WIX ファイルには問題がないにもかかわらず，原因一つに対して大量のミスマッチが生じる危険がある．この問題を防ぐために，UNIFY 演算を行う WIX ファイルは本評価実験のように見出し語に客観的に十分な識別性があることが要求される．

7. 結論，今後の課題

7.1 結　　論

本論文では WIX ファイルの統合管理を目指し研究を行った．各々独立したものとして扱っていた WIX ファイルに対し階層構造を構成することにより，WIX ファイルの分割管理を実現し，保守管理を容易なものとした．さらに，WIX ファイル結合の際に生じる重複エントリという問題に対し種々の結合演算を定義することにより問題の解決を行った上，既存のエントリから新たなエントリを生み出す可能性を示した．

7.2 今後の課題

本論文における階層構造の構築はローカルサーバーにおけるものであったが，異なるサーバー上の WIX ファイルについても対応できるようにしていきたい．また，UNIFY 演算におけるノイズについての研究，特に重複参照問題に対する研究を進めていきたい．

文　　献

[1] 高橋　健太郎，遠山　元道，“ SuperSQL による A-doc ファイルの生成 ”，データ工学ワークショップ，*DEWS2007*，2007

[2] 佐藤　裕紀，遠山　元道，“ A-doc ファイルのアタッチの機能を持つ専用ブラウザの試作 ”，データ工学ワークショップ，*DEWS2007*，2007

[3] 朱　成敏，遠山　元道，“ A-doc に基づく閲覧者主導のハイパーリンク支援 ”，データ工学ワークショップ，*DEWS2008*，2008

[4] 市東　隼，遠山　元道“ A-doc に基づく WEB リンク集再利用化ツールの試作 ”，データ工学ワークショップ，*DEWS2008*，2008

[5] 埼玉西武ライオンズオフィシャルサイト: http://www.seibulions.jp/

[6] オリックス・バファローズ: http://www.buffaloes.co.jp/

[7] 北海道日本ハムファイターズ: http://www.fighters.co.jp/

[8] 千葉ロッテマリーンズ･オフィシャルサイト: http://www.marines.co.jp/

[9] 東北楽天ゴールデンイーグルス・オフィシャルサイト: http://www.rakuteneagles.jp/

[10] 福岡ソフトバンクホークス・オフィシャルサイト: http://www.softbankhawks.co.jp/

[11] Yomiuri Giants Official Web Site: http://www.giants.jp/

[12] 阪神タイガース公式サイト: http://hanshintigers.jp/

[13] 中日ドラゴンズ 公式サイト: http://dragons.jp/

[14] 広島東洋カープ: http://www.carp.co.jp/

[15] 東京ヤクルトスワローズ: http://www.yakult-swallows.co.jp/

[16] 横浜ベイスターズ: http://www.baystars.co.jp/

[17] フリー百科事典『ウィキペディア（Wikipedia）』: http://wikipedia.org/

[18] Yahoo! カテゴリ: http://dir.yahoo.co.jp/