

WIX システムにおけるコンテンツ作成支援システム

藤井 洋太郎[†] 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: [†]yotaro@db.ics.keio.ac.jp, ^{††}toyama@ics.keio.ac.jp

あらまし 著者らは, Web ドキュメント中に存在する単語に対してリンクを生成する Web Index(WIX) システムを開発している. リンク生成には, キーワードとそれに対応する URL のエントリが記述された WIX ファイルを事前に作成する必要がある. 本稿では, XPath 式を利用し閲覧 Web ページのエントリ抽出・ファイル自動生成を可能にするウェブブラウザ拡張機能と, オンラインでのファイル作成・編集を可能にするファイル編集システムの 2 つのアプローチを提案し, 実装した.

キーワード Web Index, Web 情報システム, Web, ハイパーリンク, XPath

Content Creation Support System of WIX

Yotaro FUJII[†] and Motomichi TOYAMA^{††}

^{†††}Department of Information and Computer Science,
Keio University

Hiyoshi 3-14-1, Kouhoku-ku, Yokohama-shi, Kanagawa, 223-8522 Japan

E-mail: [†]yotaro@db.ics.keio.ac.jp, ^{††}toyama@ics.keio.ac.jp

1. はじめに

Web における利用者主導による情報資源結合を実現するために, 著者らは Web Index(WIX) と呼ぶ情報資源表現形式の提案, 開発を行っている.

現在の Web では, Web 文書作成者によって特定のアンカーテキストから特定のページへのリンクが関連付けられるという構造が一般的である. WIX では, アンカーテキストとリンクを Web ページから独立した「キーワードとリンク先の集合」(WIX ファイル) として扱い, 任意のドキュメントに対してユーザ主導で「結合」することでドキュメント内のキーワードを対応する URL のハイパーリンクに自動的に変換する.

そこで本論文では, 2 つのアプローチによる WIX ファイルの作成支援システムを提案し実装した.

本論文の構成は以下の通りである. まず, 2 章で本論文の研究目的について述べる. 3 章で WIX システムの概要を説明する. 4 章 5 章で 2 つのアプローチによる提案システムについて説明する. 6 章 7 章で評価・まとめを行う.

2. 研究の目的

WIX ファイルの作成には, 手動による記述, Web ページのリンク集からの自動生成などが挙げられる. 手動での記述の場合, ユーザーの意図が最も反映された WIX ファイルが作成できる

が, 手間と時間のコストが大きい. 一方, Web ページのリンク集からの自動生成も考えられるが, Web ページ作成やプログラミングの知識がないユーザーにとっては敷居が高い方法である. また, 作成済みの WIX ファイルに対して, リンク先の変更や追加などの WIX ファイルの情報の更新が必要になった場合, 即座に反映することができない. そこで本論文では, 一般的なユーザーでもリンク集を簡単に発見でき, そこから WIX ファイルの自動生成を可能にするシステムと, オンラインで WIX ファイルの作成・編集を支援し情報の変更に対応できるシステムが必要と考え, 以上の 2 つのアプローチによる WIX ファイル生成の効率化を目的とした.

3. Web Index

3.1 WIX ファイルの書式

WIX ファイルは XML 形式で記述されたエントリの集合であり, 記述例は図 1 のようになる. エントリ内には, キーワードとなる見出し語を keyword 要素に格納し, それに対応する詳細情報を示す参照先の URL を target 要素に格納する. また, WIX ファイルには, header 要素にファイル概要, 作者コメントなど, その WIX ファイル全体についてのメタデータを格納することも可能である.

3.2 アタッチ

閲覧中の Web 文書中にある単語について詳細な情報を得た

～企業製品 WIX の例～

```
<wix>
  <header>
    <description> アップル製品情報 </description>
  </header>
  <entry>
    <keyword>iPhone</keyword>
    <target>http://www.apple.com/jp/iphone/</target>
  </entry>
  <entry>
    <keyword>iPad</keyword>
    <target>http://www.apple.com/jp/ipad/</target>
  </entry>
  ...
</wix>
```

図1 WIX ファイルの記述例

い場合、従来ならその単語を1つずつ検索エンジンなどで検索しなければならない。これに対し WIX では閲覧中の Web 文書に WIX ファイルを結合することで、キーワードからそのキーワードの情報へのハイパーリンクを生成し、Web サイトを動的に結合することが可能である。この結合操作を“アタッチ”と呼ぶ。また、同じキーワードでもリンク先の異なる WIX ファイルを複数提供している場合、利用者は状況に応じた WIX ファイルを選択・アタッチすることで、より目的に適した情報を得ることが可能となっている。

3.3 WIX ファイルの性質

あるキーワードに対してターゲットになりうる URL は Web 上に複数存在する。例えば、キーワードがサッカー選手や芸能人の氏名である場合、ターゲットとして公式サイト、ブログ、Wikipedia などが候補として挙げられる。ターゲットは WIX ファイル作成者が WIX ファイルの目的に適したものを選択し記述する。ここで、一つの WIX ファイル内のキーワードとターゲット先は共通点があるものを記述する必要がある。仮に、キーワードによってターゲット先が Wikipedia であったり公式サイトであったりと異なるカテゴリ先が記述された WIX ファイルの場合、利用者の目的に応じたアタッチは困難となる。そのため、図2の右側の例のように同一ファイルの WIX ファイルでは共通点が多いターゲットを選びカテゴリを統一することが求められる。

キーワード：本田圭佑 ターゲット先：公式サイト	キーワード：本田圭佑 ターゲット先：Wikipedia
キーワード：長友佑都 ターゲット先：公式ブログ	キーワード：長友佑都 ターゲット先：Wikipedia

不適切な WIX ファイル例

適切な WIX ファイル例

図2 適切な WIX ファイル

4. WIX File Extractor：ウェブブラウザ拡張機能を用いた WIX ファイル生成

4.1 システム概要

本システムでは、ウェブブラウザの拡張機能を用いて、Web ページのリンク集から WIX エントリーを抽出し、自動的に WIX ファイルを作成することを可能にする。ユーザーは目的のリンク集がある Web ページを表示し、ページ上をマウスオーバーすることで、視覚的にリンク集を発見できる。また、目的のリンク集が見つかったときは、ボタンひとつで WIX ファイルを生成・確認することができる。以下の図3はこのシステムを用いてサッカーチームの選手一覧ページから、WIX ファイルを生成する例である。



図3 ウェブブラウザの拡張機能を用いた WIX ファイル生成

4.2 システム外部仕様

本システムの利用の流れは以下の通りである。主に以下の4つのフェーズを行うことで、WIX ファイルを生成できる。

(1)WEB ページ上をマウスオーバー。

目的のリンク集がある Web サイト上を読み込み、ページ上をマウスオーバーする。

(2) 目的のリンク集を探す。

目的のリンク集が選択される場所を探す。

(3) 右クリックからファイル生成ボタンを押す。

目的のリンク集が選択されたら、右クリックメニューから生成ボタンを押す。

(4) 表示された結果を確認。

また、オプションの機能として、選択範囲の広狭や取得するキーワード内の空白の有無の指定も可能である。

4.3 システムの内部仕様

本システムは、ウェブブラウザ (Firefox, safari, Google Chrome) の拡張機能として実装した。本研究では目的のページからリンク集を認識するために XPath 式を利用する。ユーザーがマウスオーバーしているノードを取得し、HTML ソースの body ノードまでの XPath 式を作成することで、ユーザーに対象ノードと同様の構造をとっている兄弟ノードの集合をユーザーにフィードバックを与えることができる。処理の流れは以下の通りである。

4.3.1 XPath 式の作成

システムはユーザのマウス操作を監視し、マウスオーバーされているノードを取得する。取得したノードから body ノードまでの XPath 式を作成する。XPath 式を作成する際、それぞれのノードの class 属性、id 属性も同時に取得する。XPath 式にそれ

らの情報を指定することで、ノードまでのより正確な XPath 式を作成できノイズの除去ができると期待できる。一方、目的のリンク集を指定できない場合は、XPath 式から属性の指定を除き、範囲の指定を緩めることができる。以下の図 4 は、図 3 においてマウスオーバーされたノードから得られる XPath 式の例である。

```
//BODY/DIV[@class="container"]//DIV[@class="innerBox"]//DIV[@class="mainColumn"]//DIV[@class="mainContentColumn"]//TABLE[@class="mainContentTbl mgB20 teamList"]//TBODY/TR/TD/UL[@class="mainContentList4"]//LI/A
```

図 4 図 3 でマウスオーバーされているノードから body ノードまでの XPath 式

4.3.2 ノードの色付け

作成した XPath 式に適合する全ての兄弟ノードを色付けする。

4.3.3 エントリーの抽出

ファイル生成ボタンが押されると、その XPath 式からキーワードとターゲットの抽出を行う。

(1) キーワードの抽出 キーワードはアンカータグに囲まれたアンカータグから取得する。しかし、リンクがテキストでなく画像である場合、アンカーテキストが存在しない可能性もあり得る。そこでリンクが画像の場合は、IMG タグ内の alt 属性から取得する。alt 属性がない場合、同タグ内の title 属性からキーワードを取得する。また、ユーザーの設定により、キーワード内のスペースの有無を指定できる。スペースを削除する場合、得られたテキスト内の空白部を削除する。

(2) ターゲット URL の抽出

ターゲットはアンカータグの href 属性から取得している。

4.3.4 抽出結果の表示

得られたキーワードとターゲットを、3.1 の WIX ファイルの書式に整形し、ユーザーが閲覧しているページ上に WIX ファイルの内容を表示する。

5. Online WIX File Editor : オンラインでの WIX ファイル作成・編集

5.1 背景

これまでの WIX システムでは、WIX ファイルの作成・編集はファイル作成者がそれぞれのローカル上でファイルを作成し、システムにアップロードするという方法をとっていた。しかし、WIX ファイル内のターゲット先の変更や、エントリーの追加や削除の更新を行うたびにファイルを編集し、アップロードするのは、非常に高コストである。また、一人の作成者によって、数百～数千のエントリーを手動で記述することは現実的に難しいと考えられる。

そこで、WIX ファイル作成支援の 2 つめの方法として、WIX ファイルをオンラインで編集可能にするシステムを提案する。

5.2 システム概要

本稿では作成者が簡単に WIX ファイルを閲覧でき、エントリーの追加・削除などの更新を行えるオンライン編集システムの構築を行った。本システムでは、主に以下の作成支援機能を実装した。

- 新規エントリーの追加・削除

- WIX ファイルのヘッダー情報追加
- キーワードに対するターゲットの推薦
- WIX ファイル内エントリーの検索
- 編集履歴閲覧・編集取消
- 登録エントリーのリンク切れチェック
- 新規 WIX ファイル作成

実際にはさらに無数の支援が考えられるが、今回はこれらについての提案をおこなう。

5.3 システム外部仕様

図 5 に、以下の機能を実装したシステムの外観を示す。本システムの利用の流れは以下の通りである。

(1)WIX ファイルの選択

WIX ファイル一覧から、閲覧・編集したいファイルを選択。

(2)WIX ファイルの編集

エントリーの追加や、ヘッダー情報は所定のフォームに入力する。またエントリー追加の際、キーワードを入力するとそのキーワードに対するターゲットの候補が自動的に表示される。ユーザーはその候補を選択することでも、ターゲットの入力ができる。エントリーの削除は、ページ左のエントリー欄のチェックボタンを入力することで削除できる。

(3) エントリーの検索

エントリー数が多く、目的のエントリーを探索しない場合は検索フォームにクエリを入力することで、適合する全てのエントリーを表示できる。検索結果はクエリを入力するたびに表示されるようになっている。図 6 にエントリー検索機能を使った例を示す。

(4) 履歴の閲覧・編集取り消し

全ての編集は履歴として保存され、編集の取り消しが可能となっている。

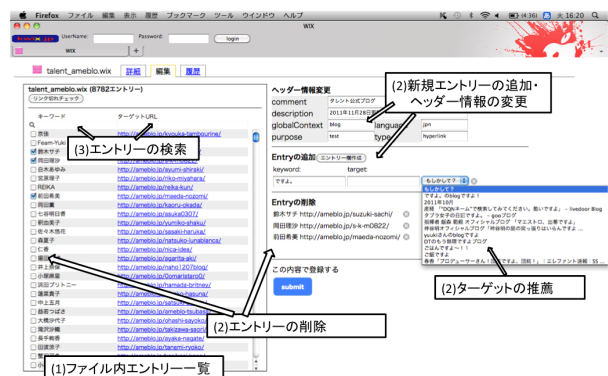


図 5 WIX ファイル編集システムの外観

5.4 システムの内部仕様

5.4.1 WIX ファイルのロード

XML 文書を読み込み操作する方法として、主に DOM を利用する方法と SAX を利用する方法の 2 種類の方法が挙げられるが、WIX ファイルのエントリー数は対象が Wikipedia などであると数 10 万件にも及ぶ可能性がある。そこで、メモリの消費量が少なく、高速に文書を走査できる SAX パーサーを利用し WIX

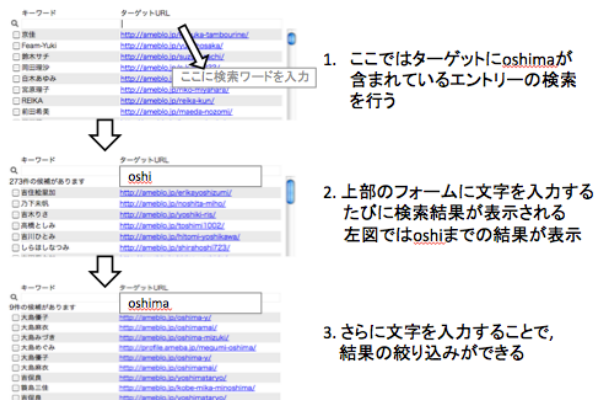


図6 エントリー検索の動作例

ファイルをロードする。

5.4.2 エントリーの逐次検索

検索フォームにキーが押されるたびに、システムは Ajax 通信を行いファイル内のエントリーと入力クエリ内容が部分一致するエントリーを検索し検索結果を表示する。Ajax 通信を用いた非同期通信を行うことでページ遷移をすることなく、動的に検索結果の書き換えを行う。

5.4.3 ターゲットの推薦

エントリー追加フォームにキーワードが入力されると、バックグラウンド上でそのキーワードに対する Google 検索を行う。得られた検索結果の HTML ソースを解析し、タイトルと URL を取得し、その上位 10 件の結果を提示する。また WIX ファイルのヘッダー情報には、WIX ファイル全体についてのコンテキストを格納できるグローバルコンテキスト属性 (ex. サッカー WIX であれば、サッカー、公式ブログ WIX であれば、ブログ等) があり、WIX ファイルがその属性を持っている場合、キーワードとグローバルコンテキストの AND 検索を行った結果を提示する。グローバルコンテキストを加えることで、より作成者の目的にあったターゲットを提示できる。提示した結果が選択されると、その URL をターゲットフォームに自動的に補完する処理を行う。

5.4.4 編集履歴

編集が行われる際、以前のバージョンとの差分を取得する。またその差分を用いることで、以前の版に戻すことが可能である。差分の取得と適用は、diff と patch コマンドといった OS レベルでの実装を行なっている。

6. 評価

以上の 2 つのシステムの有効性を評価するため、各システムを利用したユーザー実験を行い、ファイル作成時間と精度について評価する。実験は 3 名の HTML や XML を記述した経験のある被験者 (以下、経験者) と、3 名の HTML を記述した経験のない被験者 (以下、未経験者) の 6 名に参加してもらい、以下の 2 つの実験を行い評価を行った。

6.1 WIX File Extractor の評価

6.1.1 評価方法

本システムの有用性を評価するため、本システムを利用した場合と利用しない場合の WIX ファイルの作成時間、タイピング数

(キーボードタイプ数、クリック数の合計) の差を測定した。利用しない場合の時間とタイピング数は、事前に著者が 10, 20, ..., 100 件のエントリーを持つ WIX ファイルを作成し、それぞれの WIX ファイル作成にかかった時間、タイピング数の測定結果を用いた。被験者には、以下の 3 つのカテゴリの WIX ファイルを本システムを用いて作成してもらった。

- あるスポーツチームの選手情報を対象にした WIX ファイル
 - あるメーカーの製品情報を対象にした WIX ファイル
 - 被験者が興味を持っているカテゴリの WIX ファイル
- また、その後 5 分間本システムを利用して自由に WIX ファイルを作成してもらい、それらの WIX ファイルの精度 (再現率、ノイズ数) を測定した。

6.1.2 作成時間とタイピング数

図 7 に、筆者が事前にエントリー数の異なる 10 個の WIX ファイルを作成した際の作成時間と、タイピング数を示したグラフである。図 8 は、本システムを利用し作成した WIX ファイルの作成時間とタイピング数を表したグラフである。まず、図 7 のグ

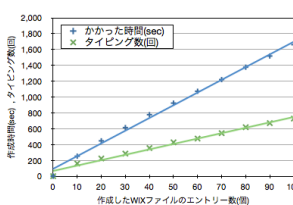


図7 本システム未使用時

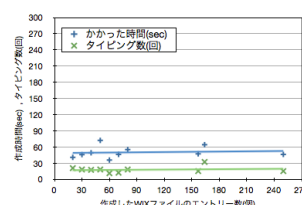


図8 本システム利用時

ラフより、作成する WIX ファイルのエントリー数が増加するにつれ、作成時間、タイピング数も線形的に増加していることが分かる。一方、本システムを利用して得られた WIX ファイルはエントリー数にかかわらず、ほぼ一定の時間とタイピング数で作成されている。100 エントリーの WIX ファイル作成において、約 1/34 の速さでの WIX ファイル生成が可能である。このことから、大規模な WIX ファイルを作成する際に本システムを利用することで、大幅なコスト削減ができると考えられる。また、図 9 に経験者と未経験者ごとの平均作成時間と平均タイピング数を示す。ここから読み取れるように両者の差は作成時間で約 5 秒、タイピング数で約 5 回と僅かなものであり、プログラミングに詳しくない一般的なユーザーでも利用可能であるといえる。

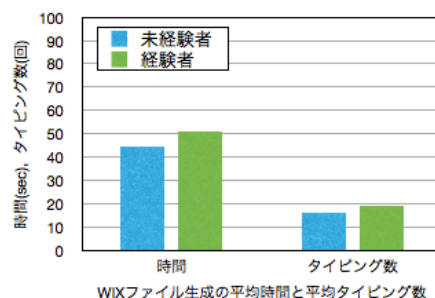


図9 経験者、未経験者の違いによる作成時間とタイピング数

6.1.3 生成 WIX ファイルの精度

次に本システムによって作成された WIX ファイルの精度の評価を行う。評価の対象となる WIX ファイルは全 36 種類であり、全ファイルにおける再現率とノイズの個数を測定した。再現率は (1) 式のように求めた。

$$\text{再現率} = \frac{\text{期待通り抽出できたエントリーの総数}}{\text{抽出したい WIX エントリーの総数}} \times 100 \quad (1)$$

表 1 は全ファイルの再現率ごとに分類した表である。この表より約 7 割の WIX ファイルで全ての目的のエントリーを得られたことがわかる。また、約 9 割の WIX ファイルが再現率 80 % 以上の割合で生成できており、本システムは非常に有用であるといえる。一方再現率が 0 % となり目的のエントリーが 1 つも得られない WIX ファイルも存在した。これらの原因として、Web サイトにおけるアンカーテキストの記述と、サイト特有の HTML 構造が原因としてあげられる。本システムは対象 Web サイトの構造に大きく依存しているため、アンカーテキストに目的以外のワードが入っていたり、全てのノードを参照してしまうような HTML 構造には対応できなくなってしまう。図 10 にエントリー数ごとの WIX ファイル内のノイズ数の分布を示しているが、エントリー数に関係なくどの規模のサイトでもノイズが含まれていることから、本システムが Web サイトの構造に大きく依存していることが分かる。今後、再現率や適合率をさらに高めるためには、アンカーテキストの部分指定や、XPath 式の範囲指定を可能にするアルゴリズムを考える必要がある。

表 1 生成した WIX ファイルの再現率ごとの分布
本システムによる再現率

再現率 (%)	0	10~60	70	80	90	100	計
ファイル数 (個)	2	0	2	1	6	25	36

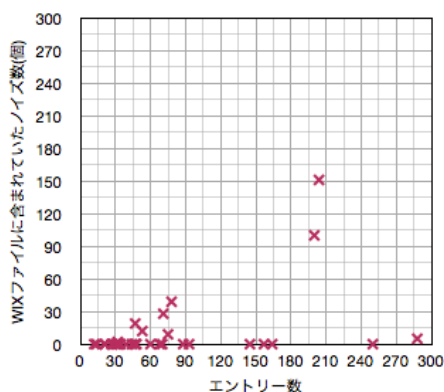


図 10 作成 WIX ファイルのノイズの分布 (エントリー数ごと)

6.2 Online WIX File Editor の評価

次に Online WIX File Editor の評価を行った。

6.2.1 評価方法

本実験では、既にシステムに登録されている WIX ファイルの

編集を行ってもらった。以下の 3 つ WIX ファイルを対象としている。

- 企業サイトへのリンクをまとめた WIX ファイル (約 100 エントリー)
- タレントブログを対象にした WIX ファイル (約 1 万エントリー)
- Wikipedia を対象にした WIX ファイル (約 10 万エントリー)

編集内容は、著者がランダムに指定した 10 個のエントリーの追加、ランダムに指定した 10 個のエントリーの削除の 2 通りを行なった。本実験は被験者にシステムを利用する場合と、利用しない場合の 2 通りの作業を行なってもらい、編集時間、タイピング数の差を測定した。本実験での編集時間とは、システムを利用する場合は本システムにアクセスした瞬間、利用しない場合はファイルのダウンロードをはじめた瞬間から、再度システムにアップロードし終わるまでの時間とした。

6.2.2 編集時間とタイピング数の考察

まず、本システムの使用有無による、編集時間とタイピング数をそれぞれ、図 11 と図 12 に示した。図 11 では縦軸に編集時間、図 12 ではタイピング数を取り、横軸は WIX ファイルのエントリー数順にとった。図 11、図 12 から読み取れるように、全ての場合において本システムのほうが良い結果となった。まず、図 11 よりシステムを利用しない場合の編集時間はエントリー数が大きくなるにつれ、大きくなっていることがわかる。この問題の大きな原因として、エントリーの増加に伴い類似するエントリーも増えてしまい検索するコストが大きくなってしまいうことがあげられる。一般的なエディタの検索機能では結果を一度に把握するのが難しいが、本システムでの検索は、1 文字ごとに検索結果が変化し直感的な操作をすることが可能である。実際、本システムでのエントリー数による編集速度の変化はほとんどなかった。一方、タイピング数はどちらも同様の推移をしていることがわかる。今回の実験において全被験者はコピー＆ペーストのショートカットキーを用いていたため、同編集数では WIX ファイルのカテゴリの影響を受けず、結果があまり変化しなかったと考えられる。しかし、本システムのターゲット推薦機能によって、余分なコピー＆ペーストを減らすことで、大幅なタイピング数の減少が可能となった。編集速度、タイピング数の両面で編集コストの削減ができ、本システムは有用性があるといえる。

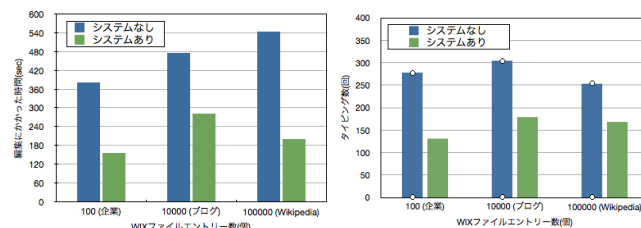


図 11 WIX ファイルの編集時間 図 12 ファイル編集時のタイピング数

6.2.3 経験の有無による違い

次に経験者と未経験者の違いによる作成時間の差を評価した。図 13 は、カテゴリごとに経験者の作成時間 (sec) 一未経験者の

の作成時間 (sec) を求めたものである。本システムを利用しない場合、エン트리数の増加に伴い経験の差が大きく現れる結果となった。本システムを利用しない場合に使うエディタやソフトウェアの指定をしていないため、操作の慣れが大きく影響していると考えられる。一方、本システムを利用した場合の差は、利用しない場合に比べ非常に小さい結果となった。本システムは、XML などの特別な知識を必要とせず、直感的な操作で編集ができるため、どのユーザーでも同程度の作成時間で編集を行えたと考えられる。

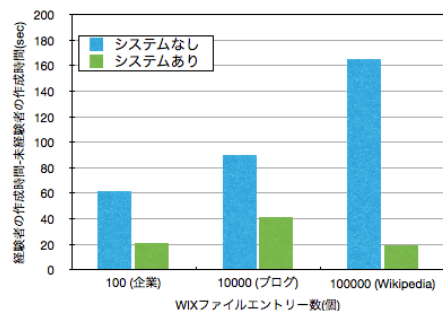


図 13 未経験者の編集時間 (タイプ数)-経験者の編集時間 (タイプ数)

7. 今後の課題

本システムを提供することで、一般的なユーザーでも簡単に WIX ファイルを作成し編集することが可能となる一方、悪意あるユーザーによるあらし行為や攻撃などが増加することが考えられる。そこで今後は、認証システムやブラックリスト、WIX ファイル・ユーザーの評価などの対策をおこない、ユーザーが安心して WIX システムを利用できる環境づくりの構築を行っていく。

8. ま と め

本論文では、2つのアプローチによる WIX ファイルの作成支援システムを提案し実装した。

1 つ目に提案した、WIX File Extractor を利用することで、作成者のファイル作成にかかるコストを大幅に減少させることができた。また、HTML やその他のプログラミング知識が無いユーザーでも目的の WIX ファイルを作成することが可能であった。どのユーザーでも簡単に素早くコンテンツ作成ができることや、複数のウェブブラウザの拡張機能として実装することで、ユーザーが普段使い慣れているブラウザ上での操作により本システムを利用することができることは大きな利点であると考えられる。

また、2 つ目に提案した Online WIX File Editor を利用することで、作成者は即座に必要な情報の更新が行なうことが可能となり、また直感的な操作により XML などの知識がないユーザーでも簡単に目的の WIX ファイルの作成・編集が可能であった。また、オンライン上に WIX ファイルを公開することで、一つの WIX ファイルを複数のユーザーで編集することが可能になり、WIX ファイル作成の効率化も期待できる。

以上の 2 つのシステムを利用することで、WIX ファイルのさらなる増加や WIX ファイルを最新の状態で提供できることが期待できる。

文 献

- [1] 山本 岳洋, 中村 聡史, 田中 克己. RerankEverything: ランキング結果閲覧のための柔軟な再ランキングインタフェース, TOD, Vol.3, No.4, pp. 48-64, 2010.
- [2] 森 良介, 藪 達也, 朱 成敏, 遠山元道. Keio WIX システム (2) サーバサイド実装, DEIM2011, 2011.
- [3] 市東 隼, 分部 亮太, 朱 成敏, 遠山元道. Keio WIX システム (3) コンテンツ作成, DEIM2011, 2011.