

複数のモバイルユーザのための Web ページ分割を用いた 協調 Web ブラウジングシステム

前川 卓也[†] 原 隆浩[†] 西尾 章治郎[†]

[†]大阪大学大学院情報科学研究科マルチメディア工学専攻
{t.maekawa,hara,nishio}@ist.osaka-u.ac.jp

筆者らはこれまでに、複数のモバイルユーザのための協調 Web ブラウジングシステムを開発している。協調 Web ブラウジングシステムは、筆者らが提案したコンテンツ記述方式で記述されたメタデータをもつ Web ページを分割し、分割したページを互いに近くにいる複数のモバイルユーザに割り当てる。これにより、ユーザは 1 つのモバイル端末だけでは閲覧できない情報量の多いページを、協調して閲覧できる。本稿では、これまでに開発したシステムにおいて、通常の HTML 形式の Web ページからメタデータを自動的に生成し、ページを分割する機構を実現する。またこれまでのシステムでは、ユーザが分割前のページの構造を把握するために、ページのオーバービューをテキストベースで提供していた。本稿では、元のページのスクリーンショット画像を用いてイメージベースのオーバービューを提供する機構を実現する。キーワード モバイルコンピューティング、コンテンツ閲覧、Web ページ分割、協調ブラウジング

A Collaborative Web Browsing System for Multiple Mobile Users by Using Web Page Splitting

Takuya MAEKAWA[†] Takahiro HARA[†] Shojiro NISHIO[†]

[†]Dept. of Multimedia Eng., Grad. Sch. of Information Science and Technology, Osaka Univ.

In our previous work, we have developed a collaborative web browsing system for multiple mobile users. The collaborative web browsing system divides a web page in which the metadata are written by our proposed content description method and allocates the divided partial pages to multiple mobile users. By doing so, users can collaboratively browse web pages of large volume, e.g., those designed for desktop PC. In this paper, we enhance the developed system by implementing a mechanism to automatically generate the metadata from conventional HTML pages. In addition, the enhanced system provides image-based overviews made from snapshots of the original web page while the previous system provides only text-based overviews for users to grasp the structure of the page.

Keywords Mobile computing, Content browsing, Web page splitting, Collaborative browsing

1 はじめに

近年、通信技術や計算機技術の発展とともに、携帯電話や PDA(Personal Digital Assistant) などのモバイル端末が身近なものとなってきている。ユーザは、モバイル端末を用いて、インターネット上のコンテンツへのアクセスや、さまざまなアプリケーションの実行をいつでもどこでも行うことができる。

しかし、依然としてモバイル端末はディスプレイサイズや計算能力に制限があるため、情報量の多いコンテンツを快適に閲覧することは困難である。そこで筆者らは、ユーザが自身のモバイル端末と、周辺にいるユーザがもつモバイル端末とを組み合わせることで、情報量の多いコンテンツを閲覧する協調ブラウジングを提案している [4]。協

調ブラウジングでは、1 つのコンテンツを複数の部分コンテンツに分割し、それぞれを各端末に割り当てることにより、1 つの端末では閲覧できない情報量の多いコンテンツを複数の端末を用いて閲覧することができる。

モバイルコンピューティング環境において、あるユーザが協調ブラウジングを行いたいときに、ユーザの身の回りある端末の数や性能などが一定であるとは限らない。また、1 つのコンテンツを複数に分割するため、それぞれの端末に割り当てられたコンテンツは関連性のあるコンテンツから構成される方がユーザにとって閲覧しやすい。しかし、このようなさまざまな条件に応じて柔軟にコンテンツを分割することは困難である。そこで筆者らは、協調ブラウジングにおける柔軟なコンテンツ分割を実現するための、コンテンツ記述方式と分割方式

を提案している [4]。コンテンツ記述方式では、コンテンツのメタデータを木構造で記述する。具体的には、コンテンツを構成する最小単位であるオブジェクトを木構造の葉ノード (オブジェクトノード) とし、関連性のあるオブジェクト同士を葉ノード以外のノード (分類ノード) で階層的に分類する。また葉ノードに、その葉ノードが表すオブジェクトのデータサイズや、オブジェクトの実行に必要なモバイル端末の機能などの属性を付加する。コンテンツ分割方式では、記述方式で記述されたメタデータ木を完全グラフに変換することによりグラフ分割問題 [3] に帰着させ、端末の数や性能などの条件に応じてコンテンツを分割する。

さらに筆者らは、協調ブラウジングの実現例として協調 Web ブラウジングシステムを開発している [5]。開発したシステムでは、文献 [4] の記述方式で記述したメタデータをもつ Web ページを分割し、複数の端末に割り当てる。ユーザは、各端末に表示された部分ページを互いに参照しながら、目的とする情報を相談して検索、閲覧することができる。

しかし、これまでに開発したシステムでは、文献 [4] の記述方式で記述したメタデータをもつ Web ページしか閲覧することができない。そこで本稿では、HTML(Hyper Text Markup Language) で記述された Web ページの構造解析を行いメタデータを自動生成することにより、通常の Web ページに対応できるように、協調 Web ブラウジングシステムを拡張する。またこれまでのシステムでは、分割されたページの構造をユーザが把握するために、元のページのオーバービューをユーザに提供していた。このオーバービューは、ページの構造をテキストベースで簡単に表したものである。本稿の拡張システムでは、ページのスクリーンショット画像を用いたイメージベースのオーバービューを作成し、ユーザに提供する。

以下、2 章では協調ブラウジングについて述べ、3 章では拡張した協調 Web ブラウジングシステムについて述べる。最後に、4 章で本稿のまとめとする。

2 協調ブラウジング

協調ブラウジングでは、1 つの情報量の多いコンテンツが複数のコンテンツに分割される。より正確には、コンテンツを構成する複数のオブジェク

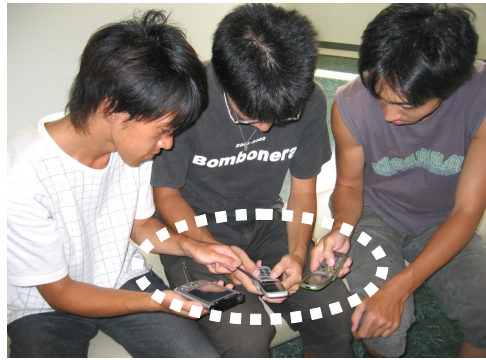


図 1: 協調ブラウジングを行っている様子

トを、分割数に応じてオブジェクト集合に分ける。各端末には、分割されたオブジェクト集合が配布される。ユーザは、各端末に表示されたコンテンツを互いに参照し、協調してコンテンツの閲覧を行う。図 1 に協調ブラウジングを行っている様子を示す。

以下では、筆者らが文献 [4] で提案した協調ブラウジングのためのコンテンツ記述方式と分割方式の概要を説明する。

2.1 コンテンツ記述方式

コンテンツ記述方式は、コンテンツ分割に必要なメタデータを XML(eXtensible Markup Language) 形式の木構造で記述する。図 2 に示す A~F のオブジェクトからなるコンテンツの木構造の例を図 3 に示す。葉ノード (オブジェクトノード) は、コンテンツを構成する最小単位であるオブジェクトを表す。葉ノード以外のノード (分類ノード) は、関連性のあるオブジェクトノード同士を階層的にグループ化するものであり、コンテンツの分割に利用する。記述方式では、オブジェクトノード同士のパスの距離が小さいほど関連性が大きいと定義している。例えば図 3 において、ノード G1 の子であるノード A と B の距離は 2 となり、ノード A と D の距離は 4 となるため、ノード A はノード D より B に関連性があると言える。さらに記述方式では、オブジェクトノードに対して以下のような属性を付加できる。

オブジェクト名 オブジェクトが表すコンテンツの名前を表す。

オブジェクトの ID メタデータ木におけるオブジェクトの識別子を表す。

オブジェクトの量 実際のオブジェクトのデータサ

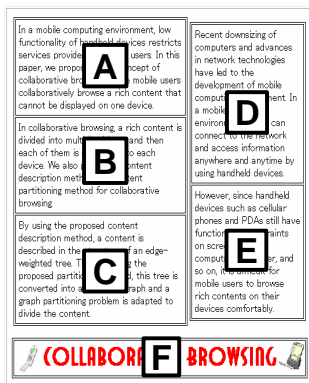


図 2: コンテンツの例

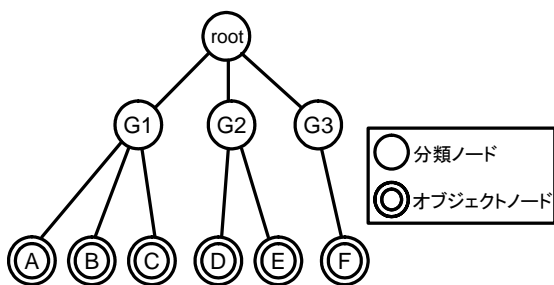


図 3: コンテンツ構造の例

サイズなど、オブジェクトの量の指標となるものを表す。この値はオブジェクトノード自体の重みと対応付ける。

実行に必要な端末の機能 オブジェクトを実行するために、端末に必要な機能を表す。

2.2 コンテンツ分割方式

コンテンツ分割方式は、記述方式で記述したメタデータ木を完全グラフに変換し、グラフを端末と同じ数の部分グラフに分割する。このときグラフは木のオブジェクトノードのみからなり、ノード間の枝の距離が木におけるノード間のパスの距離に対応するように変換する。分割の際、以下に示す条件を考慮することができる。

端末に割り当て可能なコンテンツの量 端末によっては利用可能なメモリやストレージの容量が制限される。例えば、携帯電話には表示可能な Web ページの最大容量が制限されている。そのため、各端末の最大容量を超えないようにコンテンツを割り当てる。

端末の性能 端末によっては画面の大きさや通信速度などが制限される。そのため、コンテンツを複

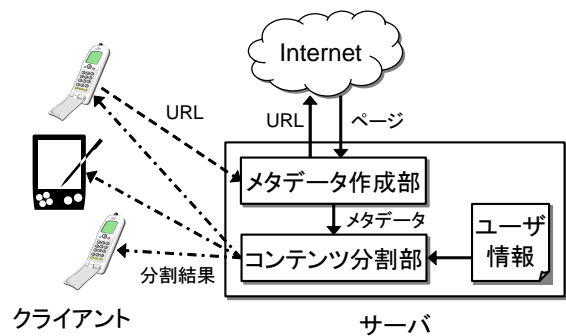


図 4: システム構成図

数の端末に分配するとき、各端末の性能比に応じてコンテンツの分配割合を決定する。

端末の機能 実行に特別な機能が必要なコンテンツ (オブジェクト) は、その機能をもつ端末に割り当てる必要がある。例えば、GIF 形式の画像を表示できない端末には、GIF 形式の画像を割り当てない。

分割したコンテンツの関連性 分割されたコンテンツを構成するそれぞれのオブジェクト集合は、関連性のあるオブジェクトから構成されている方が閲覧しやすい。つまり、グラフにおいて枝の距離が近いノード同士を同じ部分グラフとなるようにグラフを分割する。

ユーザの嗜好 ユーザに嗜好が存在する場合、ユーザの嗜好に沿ってコンテンツ (オブジェクト) を割り当てることが望ましい。例えば、野球に興味のあるユーザには、“野球” という語を含むコンテンツを割り当てる。

3 協調 WWW ブラウジングシステム

本章では、本研究で拡張した協調 WWW ブラウジングシステムについて述べる。

3.1 システム概要

システムの概要を図 4 に示す。クライアントがサーバに URL を送信すると、サーバ上のメタデータ作成部は対応する Web ページの解析を行い、2.1 節で述べた木構造のメタデータを作成する。コンテンツ分割部は、作成したメタデータとユーザ情報から 2.2 節で述べたコンテンツの分割を行い、分割結果をクライアントに送信する。クライアント

```

<html>
<head> ... </head>
<body>
  <table border="1">
    <tr>
      <td>
        <table border="1" bordercolor="BLACK">
          <tr> <td> <cbrowsing:node-content id="A"/>
          </tr>
          <tr> <td> <cbrowsing:node-content id="B"/>
          </tr>
          :
        </table>
      </td>
    </tr>
    :
  </table>
</body>
</html>

```

図 5: レイアウトデータの例

は受信した分割結果から、自身に割り当てられた部分ページをブラウザ上に表示する。ここでユーザ情報は、クライアントがサーバにあらかじめ送信するものであり、端末に割り当て可能なコンテンツの最大バイトサイズ、端末の画面のサイズ、端末が表示可能な画像のフォーマット、および、ユーザの嗜好を反映したキーワード群が含まれる。キーワードはユーザがあらかじめ入力したものである。また分割結果は、各オブジェクトがどの端末に割り当てられたかの情報、Web ページ内のオブジェクトの配置を示すレイアウトデータ、および、自身の端末に割り当てられたオブジェクト (HTML コンテンツ) の集合からなる。

レイアウトデータは、Web ページから全てのオブジェクトに対応する部分 HTML テキストを取り除き、そこにオブジェクトの識別子を埋め込んで作成したものである。レイアウトデータの例を図 5 に示す。レイアウトデータ内の <cbrowsing:node-content> はオブジェクトの識別子を示す。この例では、7 行目の <table> 内のセルに id="A" と id="B" のオブジェクトが配置されることを示している。

本システムでは、ユーザがブラウザに表示されたリンクを選択すると、対応する URL がサーバに送信され、上述の処理を経て分割されたページが全てのクライアントのブラウザ上に表示される。つまり全てのクライアントにおいて、リンクの選択操作が同期している。さらに、ブラウザのバック、フォワード操作も同期しており、例えばあるユーザ

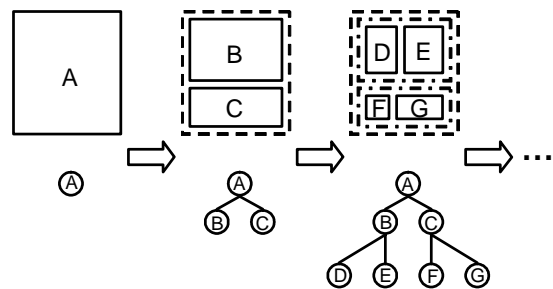


図 6: ページのセグメント化

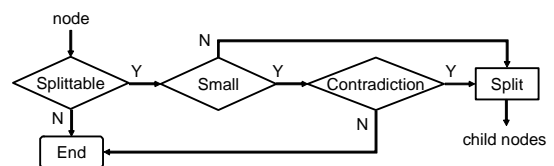


図 7: ノード分割のプロセス

がバック操作を行うと、他の端末でも同期してバック操作が行われる。

3.2 メタデータの生成

本システムでは、サーバが Web ページをセグメント化することによりオブジェクトを抽出し、オブジェクトを葉ノードとした木を作成する。ここで、Web ページをセグメント化する研究は、これまでに多く存在する [1][2][6]。本システムでは、一般的に用いられている単純な方法でセグメント化を行う。ただし、協調ブラウジングのためのセグメント化方式では、いくつかの制限を考慮する必要がある。例えば、GIF 画像のみを表示できる端末と PNG 画像のみを表示できる端末があるとすると、それらの端末は GIF 画像と PNG 画像を両方含むオブジェクトを実行できない。また端末は、その容量を超えたサイズのオブジェクトを実行できない。

本システムのセグメント化方式では、最初にページ全体を木のルートノードに設定する。その後、図 6 に示すように、葉ノードに対応するオブジェクト (HTML ページ) をさらに複数のページに分割し、分割されたページをそのノードの子ノードに設定する。この処理は葉ノードが分割可能な限り再帰的に繰り返される。このようにページをセグメント化し木を作成することにより、ページ上で近くにレイアウトされたオブジェクト同士は、木においてパスの距離が短い位置に配置される。例えば図 6 の右端の図において、ページ上で隣接するノード

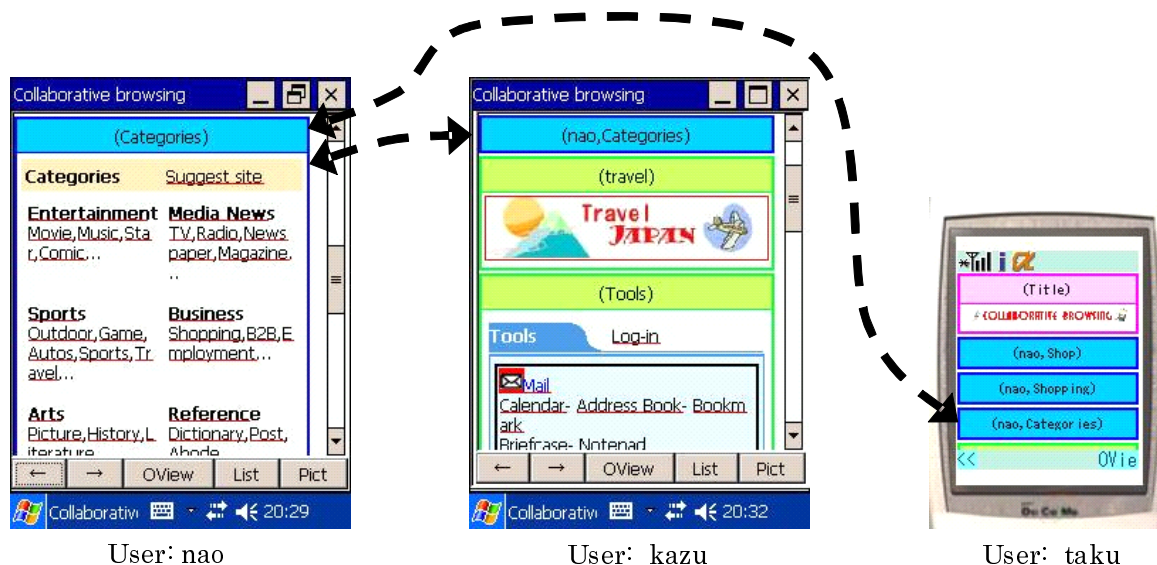


図 8: ブラウザ画面の例

D と E のパスの距離は 2 となるが、ノード D と F の距離は 4 となる。このようにして作成した木を、文献 [4] の分割方式を用いて分割することにより、関連性の大きいオブジェクト (ノード) 同士を同じクライアントに割り当てることができる。

図 7 に、1 つのノードを複数の子ノードに分割するプロセスを示す。このプロセスでは最初に、特定の HTML タグを用いてノードが分割可能か判定する。具体的には、<p>、、、<hr>、連続する
、<table>などのページセグメントに一般的に用いられるタグを用いて判定する。複数の<p>が存在する場合、<p>に括られた部分を子ノードとする。またはが存在する場合、リスト中の各アイテムを子ノードとする。<hr>または連続する
が存在する場合、それらのタグの前後の部分をそれぞれ子ノードとする。さらに<table>が存在する場合、テーブルの各セルを子ノードとする。ノードが上述のタグにより分割可能と判定された場合、そのノードのサイズを判定する。サイズが閾値より大きいとき、上述のタグを用いてノードを分割する。本システムでは、閾値として、最も小さい容量をもつ端末の容量の 1/2 の値を用いている。ここでノードのサイズとは、そのノードに対応する HTML の部分テキストのバイトサイズであり、その部分テキストがを含む場合は、対応する画像のバイトサイズも加えられる。ノードのサイズが閾値より小さいとき、そのノードが前述

の制限に矛盾していないかを判定する。矛盾している場合、矛盾が解消するまでそのノードを分割する。

以上のような手順でページから木を作成した後、葉ノード (オブジェクトノード) に以下のメタデータを付加する。

ノード名 本システムではノード名を、ノード (オブジェクト) の最初に現れるタグ以外の文字列、または、の alt 属性の値としている。

ノードのサイズ 各ノードに対応する HTML の部分テキストのバイトサイズを、そのノードのサイズとしている。その部分テキストがを含む場合は、対応する画像のバイトサイズもノードのサイズに加えられる。

実行に必要な機能 このノードがを含む場合、その画像のフォーマットを表示するための機能を、メタデータとして付加する。

本システムでは、以上のような木構造とメタデータを用いて、文献 [4] のコンテンツ分割方式により、コンテンツの分割を行う。

3.3 クライアントシステム

クライアントシステムは、サーバから受信した分割結果に基づいて、部分ページをブラウザに表示する。図 8 に、PDA2 台と携帯電話 1 台で協調ブラウジングを行っている際のブラウザ画面の一例を示す。ブラウザには、自身の端末に割り当てられたオブジェクトの内容、および、他の端末に割り当て

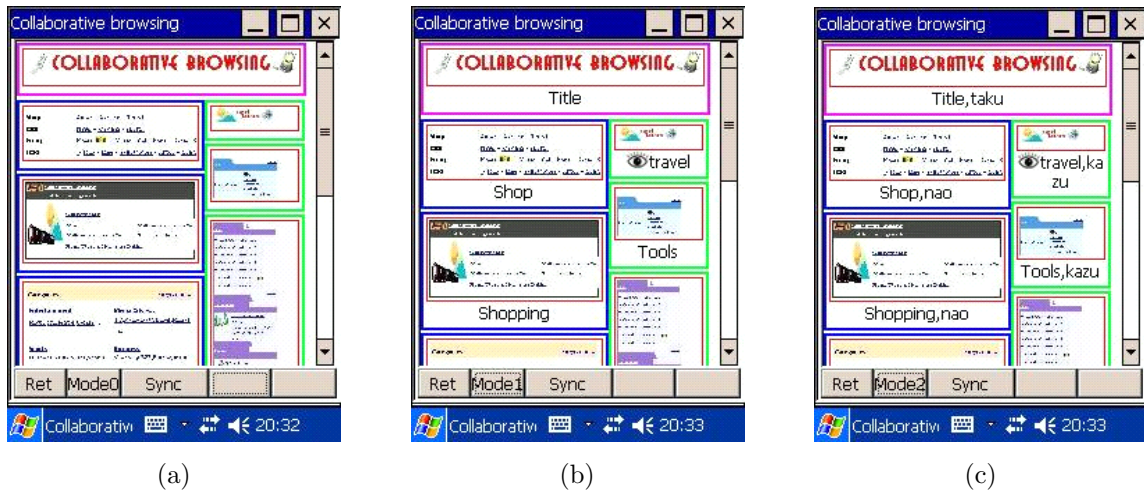


図 9: オーバービューの例

られたオブジェクトの情報をレイアウトデータに埋め込むことにより作成したページが表示される。このとき、レイアウトデータの<table>を除去することにより、オブジェクトをシリアルに表示している。また、各オブジェクトを矩形領域で囲むことにより、オブジェクト同士の境界を明確にしている。自身に割り当てられているオブジェクトを示す矩形上部には、そのオブジェクト名を示している。他の端末に割り当てられているオブジェクトを示す矩形には、オブジェクト名とそのオブジェクトが割り当てられているユーザ名を示している。各オブジェクトやオブジェクトの情報を囲う矩形は、オブジェクトが割り当てられているユーザごとに色分けして表示している。例えば、図 8 のユーザ “kazu” の PDA には、ユーザ “nao” に割り当てられているオブジェクト “Categories” を示す矩形が表示され、その下には、自身の端末に割り当てられているオブジェクト “travel” と “Tools” が表示されている。ユーザ “kazu” がオブジェクト “Categories” を見たい場合は、ブラウジング画面から “Categories” がユーザ “nao” に割り当てられていることがすぐ分かるため、ユーザ “nao” に簡単に見せてもらうことができる。

またクライアントシステムは、分割前のページのオーバービューを表示できる。文献 [5] のシステムでは、オーバービューに含まれるオブジェクトの情報として、オブジェクトの名前やオブジェクトが割り当てられているユーザ名などのテキスト情報のみを提供していた。しかし、ユーザが以前にデスクトップ PC で閲覧した経験のある Web ページ

を協調ブラウジングする場合、そのページのスクリーンショットをオーバービューとして提供した方が、探している情報がどこにあるかを容易に把握できることが多い。そこで本システムでは、サーバがオブジェクトごとのスクリーンショット画像を作成し、クライアントはそれらを受信しオーバービューを作成してユーザに提供する。具体的には、オブジェクトごとのスクリーンショット画像をレイアウトデータに埋め込むことにより、イメージベースのオーバービューを作成する。

オーバービューは、その詳細度を 3 段階で切り替えることができる。図 9(a) では、ユーザごとに色分けした枠内に、各オブジェクトのスクリーンショット画像のみを表示している。図 9(b) では、画像の下にオブジェクト名を表示しており、図 9(c) では、オブジェクトが割り当てられているユーザ名も表示している。またオブジェクトを表す画像はリンクになっており、ユーザが自身に割り当てられているオブジェクトに対応する画像を選択すると、オーバービュー画面から通常のブラウザ画面に切り替わり、対応するオブジェクトまでスクロールして表示される。自身に割り当てられていないオブジェクトに対応する画像を選択すると、そのオブジェクトが割り当てられているユーザの画面に、対応するオブジェクトを今すぐ表示するか否かを選択するダイアログが表示される。表示することを選択した場合、通常のブラウザ画面に切り替わり、対応するオブジェクトまでスクロールして表示される。さらに眼のアイコンは、オーバービューに切り替える前にブラウザで表示していたオブジェク

トを示す。

ユーザは、以上のような機能を用いて特定のオブジェクトを画面上に表示し、各端末に表示されているオブジェクト同士を比較しながら閲覧することができる。

3.4 システムの実装

協調 Web ブラウジングシステムのクライアントは、PDA と携帯電話上に Java 言語を用いて実装した。PDA としては Hewlett-Packard 社製 iPAQ Pocket PC h5500 を用いた。携帯電話としては、NTT DoCoMo i アプリ対応携帯電話、SH900i および F900i を用いた。携帯電話上に実装したクライアントには、サイズが 20KB までのページを表示できるように制限を設けており、PDA 上に実装したクライアントには制限を設けていない。また、PDA の解像度は 240×320 ドットだが、そのうち 240×292 ドットを実装したアプリケーションが利用している。SH900i および F900i の解像度はともに 240×320 ドットだが、そのうち 240×252 ドットおよび 240×240 ドットを実装したアプリケーションが利用している。サーバは Red Hat Linux 上に Java 言語および PHP (Hypertext preprocessor) 言語を用いて実装した。

4 おわりに

本稿では、筆者らが開発している協調 Web ブラウジングシステムを拡張し、通常の Web ページに対応させた。拡張したシステムでは、サーバが HTML 形式の Web ページを再帰的にセグメント化することにより、木構造のメタデータを作成する。そして、その木を分割することにより Web ページの分割を行う。クライアントは、分割された部分ページを受信し端末に表示する。またクライアントは、サーバが作成したオブジェクトごとのスクリーンショット画像を用いて、イメージベースのオーバービューを作成して表示する。これらの機能を用いることにより、ユーザは通常のデスクトップ PC 向けに作成された Web ページを、複数の携帯端末を用いて閲覧できる。

今後の予定としては、セグメント化方式の改良を検討している。本稿の拡張システムでは、オブジェクトのサイズをそのオブジェクトに対応する部分 HTML テキストのバイトサイズおよびオブジェ

クトが含む画像のバイトサイズの合計とし、そのサイズを超えないようにセグメント化している。この方式では、例えば長文のみを含むオブジェクトは、画像のみを含むオブジェクトに比べてブラウザに表示される範囲が大きいため、閲覧やスクロールに大きな時間がかかる。この問題を解決する一つの方法として、オブジェクトがブラウザに表示される範囲の大きさを基準に、Web ページをセグメント化することが考えられる。

謝辞 本研究の一部は、平成 16 年度受託研究(独立行政法人情報通信研究機構)「モバイル端末による協調ブラウジングのためのアプリケーション開発に関する研究」および文部科学省 21 世紀 COE プログラム「ネットワーク共生環境を築く情報技術の創出」の研究助成によるものである。ここに記して謝意を表す。

参考文献

- [1] Y. Chen, W. Ma, and H. Zhang, "Detecting web page structure for adaptive viewing on small form factor devices," Proc. World Wide Web Conference (WWW'03), pp. 225-233, May 2003.
- [2] D.W. Embley, Y. Jiang, and Y.-K. Ng, "Record-boundary discovery in web documents," Proc. ACM SIGMOD'99, pp. 467-478, June 1999.
- [3] B.W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," Bell System Technical Journal, Vol. 49, No. 2, pp. 291-307, Feb. 1970.
- [4] 前川卓也, 上向俊晃, 原 隆浩, 西尾章治郎, "複数のモバイル端末による協調ブラウジングのための木構造型コンテンツ記述方式と分割方式," 情報処理学会論文誌: データベース, Vol. 45, No. SIG7(TOD 22), pp. 11-23 (2004 年 6 月).
- [5] 前川卓也, 原 隆浩, 西尾章治郎, "複数のモバイルユーザによる協調的なコンテンツ閲覧が可能な WWW ブラウジングシステム," 情報処理学会研究報告(データベースシステム研究会 2004-DBS-134), Vol. 2004, No. 72, pp. 615-622 (2004 年 7 月).
- [6] B.L. Welch, "HTML page analysis based on visual cues," Proc. 6th International Conference on Document Analysis and Recognition, pp. 10-13, Sept. 2001.