

Function-Based Object Model Towards Website Adaptation

Jinlin Chen
Microsoft Research China
No. 49, Zhichun Road
Beijing, 100080, China
86-10-62617711
jlchen@microsoft.com

Baoyao Zhou
Dept. of Automation, Tsinghua Univ.
Haidian District
Beijing, 100084, China
86-10-62617711
goodbaby@263.net

Jin Shi
Microsoft Research China
No. 49, Zhichun Road
Beijing, 100080, China
86-10-62617711
killy@263.net

Hongjiang Zhang
Microsoft Research China
No. 49, Zhichun Road
Beijing, 100080, China
86-10-62617711
hjzhang@microsoft.com

Qiu Fengwu
Dept. of Automation, Tsinghua Univ.
Haidian District
Beijing, 100084, China
86-10-62782422
wqf-dau@mail.tsinghua.edu.cn

ABSTRACT

Content understanding is a crucial issue for website adaptation. In this paper we present a Function-based Object Model (FOM) that attempts to understand authors' intention by identifying Object function instead of semantic understanding. Every Object in a website serves for certain functions (Basic and Specific Function) which reflect authors' intention towards the purpose of an Object. Based on this consideration we have proposed the FOM model for website understanding. FOM includes two complementary parts: Basic FOM based on the basic functional properties of Object and Specific FOM based on the category of Object. An automatic approach to detect the functional properties and category of Object is presented for FOM generation. Two level adaptation rules (general rules and specific rules) based on FOM are combined for practical adaptation. A system for web content adaptation over Wireless Application Protocol (WAP) is developed as an application example of the proposed model. Experiments have shown satisfactory results and extensibility.

Keywords

Content Adaptation, Website Understanding, Content Function, HTML/WML Conversion

1. INTRODUCTION

The increasing diversity in terms of devices, protocols, networks and user preferences in today's web has made adaptive capability critical for Internet applications. Many efforts have addressed this problem and various solutions have been proposed [1] - [5].

To achieve adequate adaptation it is crucial to understand website structure and content function that reflect authors' intention directly. However, most of the previous works achieve adaptation only under some special conditions due to the lack of structural information. Some works tried to extract semantic structural information from HTML tag either manually [6] - [9] or automatically [10] [11]. But these approaches lack an overview of the whole website. In addition, they are only suitable for HTML contents. Furthermore, most of them do not work effectively even for HTML pages because HTML was designed for both presentational and structural representation of content. Further misuses of structural HTML tags for layout purpose make the situation even worse. Cascade Style Sheet (CSS) [12] compensates this situation by representing the presentation information separately, but its application is quite limited. Moreover, the difficulty of extracting semantic structure from HTML tags is still not solved by CSS. Therefore, the results of semantic rules based approaches for HTML contents are not so stable for general web pages. Smith et al. proposed InfoPyramid model [13] to represent the structural information of multimedia contents. However, InfoPyramid does not exist in current web contents. XML provides a semantic structural description to content by Document Type Description (DTD) [14]. However, DTD is not a general solution because each application area would require its special DTD, and XML takes no consideration to the function of content. Besides, although XSL [15] provides a flexible way of presenting the same content in different devices, it needs to be generated manually, which would be a labor-intensive work for authors.

Instead of semantic understanding, in this paper we present a Function-based Object Model (FOM) for website adaptation. Taking semantic understanding a task of XML, FOM attempts to understand authors' intention by identifying Object function and category. In FOM Objects are classified into Basic Object (BO) and Composite Object (CO). Each Object in a website serves certain functions including Basic Function and Specific Function. They both reflect authors' intention towards the purpose of an

Object, which is a crucial clue for web content adaptation. Based on this observation, we propose the FOM model for website understanding. FOM includes two complementary parts: Basic FOM and Specific FOM. Basic FOM represents an Object by its basic functional properties and Specific FOM represents an Object by its category. Combining Basic FOM and Specific FOM together, we can obtain a thorough understanding of authors' intention towards a web site.

Since authors tend not to give more description on Object function and category, in this paper an automatic approach to detect the functional properties and categories of Objects is presented for FOM generation, which is crucial for the application of FOM.

FOM provides two level guidelines for web content adaptation: general rules based on Basic FOM and specific rules based on Specific FOM. An example of web content adaptation over Wireless Application Protocol (WAP) is presented in this paper, and experiments have shown satisfactory results

In Section 2, we present the concept of FOM. This is followed by the description of automatic FOM generation scheme in Section 3. In Section 4, we present general adaptation rules based on Basic FOM and specific adaptation rules for WAP devices based on Specific FOM as an example. Conclusions are given in Section 5.

2. FUNCTION-BASED OBJECT MODEL

Object is the basic element of a hypermedia system. It is a piece or a set of information to perform some certain functions. According to the number of content bodies an Object contains, Object can be classified into Basic Object (BO) and Composite Object (CO). In the following two sub-sections we will illustrate Basic FOM and Specific FOM for Object, respectively.

2.1 Basic FOM – Functional Property of Object

2.1.1 Basic FOM of Basic Object

In a hypermedia system, a *Basic Object* is the smallest information body that cannot be further divided. Only as a whole can it perform certain functions.

1) Function of BO

A BO generally performs one or more of the following basic functions:

It may provide some semantic information to users or guide users to other Object with a hyperlink. Besides, it may also serve for beautifying a page or have interface for users to communicate with the system.

2) Property of BO

According to its function, a BO has the following properties:

Presentation: the way a BO shows itself to users, which includes Media Type, Layout Format and Encoding Format.

Semanteme: the content meaning of a BO. Since XML has a good scheme for describing the semantic meaning of contents, here semanteme is more at semantic layer instead.

Decoration: to what extent a BO serves for beautifying the web page. Decoration value of a BO is indicated as x , ($0 \leq x < 1$). The higher x is, the Object serves more for decoration. If x equals to 1, the BO serves only for decoration without any other information. If x equals to 0, the BO has no intention for decoration.

Hyperlink: the Object a BO points to, which has the following cases: No Hyperlink, Hyperlink to Other Object and Hyperlink to Other Application.

Interaction: the interaction method of a BO, which has the following cases: Display (for presenting information), Button (for selecting list item or submitting information) and Input (for inputting information).

3) FOM of BO

Based on the properties above, the FOM of a BO is as following:

BO (Presentation, Semanteme, Decoration, Hyperlink, Interaction)



Figure 1. An index page from <http://www.microsoft.com>

For example, the FOM of O6 in Figure 1 is:

Presentation:

Layout Format: Left aligned...

Media Type: Text

Encoding: English, Text/HTML...

Semanteme: Abstract

Hyperlink: NONE

Decoration: 0

Interaction: Display

2.1.2 Basic FOM of Composite Object

A *Composite Object* is a set of Objects (BO or CO) that perform some certain functions together. These Objects are combined under some basic clustering rules. Since a web page is composed of COs and BOs, and a website is a set of web pages, both of them are COs.

Unfolding all the child COs inside a CO, and repeating this process until all the leaves of the original CO are BOs, we have a tree-like representation of the CO, which is called the Full Representation of the CO. Root Child is a child that connects with the root directly. And Root Children are Brother Objects to each other.

1) Function of CO

Besides the basic functions of BOs, a CO has also clustering function. The Root Children of a CO are clustered based on some basic rules to perform certain goals that reflect authors' intention towards the relationship and hierarchy of Root Children.

2) Property of CO

Based on Clustering function, a CO has its special properties as following:

Clustering Relationship

- Complement: Root Children of a CO are complementary to achieve a goal and they have different basic properties.
- Parallel: Root Children of a CO are of equal importance to achieve a goal and they generally have the same basic properties. If the root children have both similar and different properties, their relationship will be calculated by a weighted

sum of the similarity of these properties. Then a threshold will be set to decide whether they are complementary or parallel.

- Table: Root Children of a CO can be clustered into parallel Root Children according to two semantic dimensions (normally row/column headers of a table).

Presentation Relationship: presentation order (time and space) of Root Children inside a CO, and whether the Root Children are separable when they are presented.

3) FOM of CO

Based on the properties above, the FOM of a CO is as following:

$CO = \{O_i, \text{Clustering Relationship, Presentation Relationship} \mid O_i \text{ is the Root Children of the CO, } i=1,2,\dots, NR\}$, where NR is the total number of Root Children of a CO.

For example, the FOM of O3 in Figure 1 is:

Root Children: O3-1, O3-2, O3-4

Clustering Relationship: Complement

Presentation Relationship: Spatial Order: Vertical; Separable

Figure 2 is the Full representation of O3.

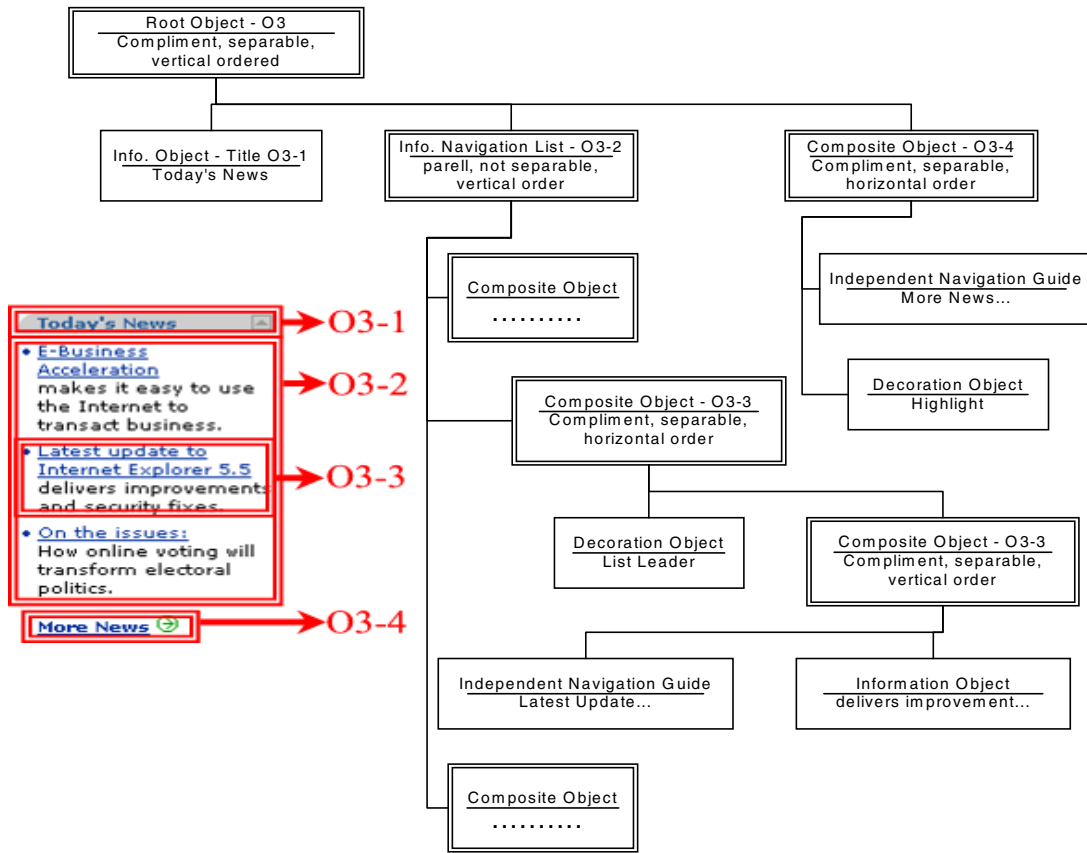


Figure 2. Root Children of O3 in Figure 1

2.2 Specific FOM – Category of Object

The specific function of an Object in a given application environment is represented by its category, which reflects authors' intention directly. There are many Objects categories according to various purposes. Here we take the HTML content of Figure 1 as an example to describe Object categories.

- 1) Information Object: presents content information. (O6)
- 2) Navigation Object: provides navigation guide. (O12)
 - Navigation Bar is a CO with parallel Navigation Objects as Root Children, and the CO exists in a set of web pages in the website to provide global navigation guide. It has the following formats: Normal Navigation Bar, Frame, Menu (O2) and Map.
 - Navigation List is a CO with parallel Navigation Objects as Root Children, and the CO exists only in a single web page to provide local navigation guide. It has the following format: Normal Navigation List (O13), Informative Navigation List (O3), Narrative Navigation List and Map List. In Informative Navigation List, each Navigation Object is followed by an Information Object as introduction. In Narrative Navigation List, all Navigation Objects are embedded in a piece of narrative information.
 - Independent Navigation Guide is an Object with hyperlink property and/or introductory information to provide navigation guide to a certain piece of content. It generally has weak relationship with other Objects around it. (O7)
- 3) Interaction Object: provides user side interaction. It has the following formats: User Selection (for selecting items from a list of available information), User Query (for inputting query information, O11) and User Submission (for uploading information).
- 4) Decoration Object: serves for decoration purpose only. (O1). Decoration Object may have the following format: Separator (O8), List Leader (O4), Highlight (O5) and Background.
- 5) Special Function Object: performs special functions such as AD, Logo, Contact, Copyright, Reference, etc.
- 6) Page Object: serves as the basic document of a website for presenting related information. It has two basic categories: Index Page and Content Page. Index Page serves mainly as navigation guide to other pages. Figure 1 is an example. The purpose of Content Page is to deliver semantic information. See Figure 11 for an example.

Note: websites can also be considered as COs. However, due to the complexity of websites, we leave the category analysis of websites for future research.

3. AUTOMATIC FOM ANALYSIS FOR HTML WEBSITES

Although it is desirable that additional information be added for the generation of FOM in the authoring phase, authors actually tend not to do so. Thus, it is important that we can automatically analyze the function of contents in a website. Here we present an automatic method for the generation of Basic and Specific FOM

in Section 3.1 and 3.2, respectively. Although our analysis is based on HTML websites, it is easy to extend to other languages.

3.1 Basic FOM Generation

3.1.1 Object Detection

Object detection is a necessary step before the generation of FOM.

In HTML content, a BO is a non-breakable element within two tags or an embedded Object. There is no other tag inside the content of a BO. According to this criteria it is easy to find out all the BOs inside a website.

COs can be detected by layout analysis of web pages. The basic idea is that Objects in the same category generally have consistent visual styles and they are separated by apparent visual boundaries from Objects in other categories. According to this observation, we first render the visual image in our system. Then a visual similarity pattern detection algorithm based on DBSCAN [16] is applied to extract Objects in the same category. The distance between different objects in this algorithm is measured by their visual similarity.

3.1.2 Basic FOM Generation for BO

The functional properties of a BO are generally included in its HTML source content. Following are some basic rules for the Basic FOM generation of BO:

- 1) Presentation Property can be determined by analyzing HTML source and tags to extract Media Type, Encoding Format and Layout information of an Object.
- 2) Semanteme Property can be determined by analyzing the content itself to extract the semantic layer.
- 3) Navigation Property is the destination of the hyperlink contained in a BO.
- 4) Decoration Property varies between [0,1] according to presentation and Semanteme property. Text/Video Objects normally have lower decoration value. And the following Objects generally have higher value to indicate that their main purpose is for decoration: general decoration symbols, lines and separators between Objects and Object with "background" property in HTML tag.
- 5) Interaction Property of a BO may be one of the following three categories:
 - Button for Object with the <button> tag and/or some button-like selection list.
 - Input Text for Object with <Input...> or related tags.
 - Display for interaction property of other Objects.

3.1.3 Basic FOM Generation for CO

Following are some basic rules for the Basic FOM generation of CO:

- 1) Clustering Relationship, may be one of the following three categories:

- Complement: the Root Children are neighborhood and have one or more different basic properties (such as Presentation or Semantic Layer).
- Parallel: the Root Children are neighborhood and have similar basic properties.
- Table: the Root Children have table tag and 2-dimensional clustering headers (column and row header).

2) Presentation Relationship

- Time Order: generally no time sequence unless required by the Object.
- Space Order can be determined by analyzing the visual image of the content.

- Root children are generally separable except special cases (such as Object for input).

3.2 Specific FOM Generation

3.2.1 Specific FOM Generation for BO

As described above, Specific FOM represents an Object with its category. For a BO, its category is mainly determined by the major properties of the BO, the properties of the Father/Brother Objects. A rule based decision tree is applied to determine the category of BO. Figure 3 illustrates the basic idea of BO category detection.

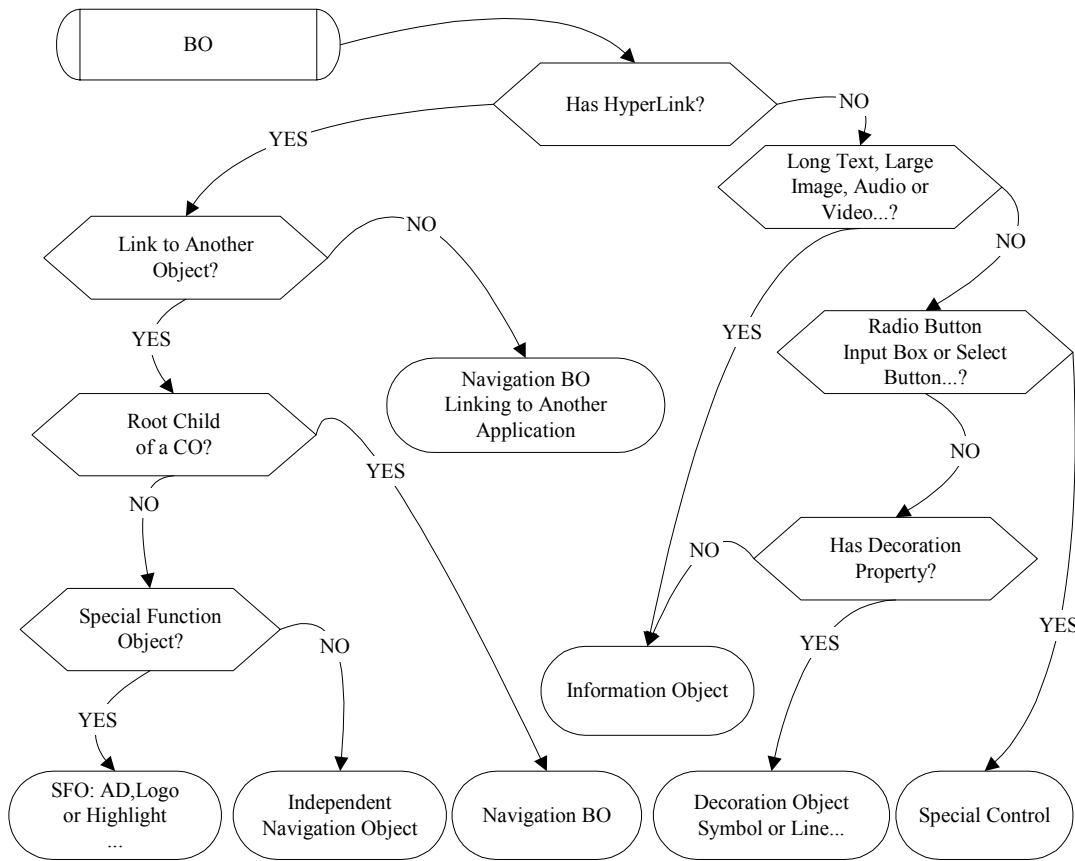


Figure 3. A basic decision tree of BO category

3.2.2 Specific FOM Generation for CO – Example 1: Navigation Bar Detection

The category of a CO is determined by the major properties of the CO/Root Children and the application environment. Each category needs a specific detection algorithm. In this section and next section we give two examples on Normal Navigation Bar (NNB) detection and page category detection, respectively. The methods can be easily extended to related categories.

According to its media type, NNB can be classified into Text and Image NNB. Here we focus on the detection of Text NNB. Image NNB can also be detected with similar method.

To find out the Text NNB in a website, we have the following rules by statistical analysis:

- Most of the Root Children of a Text NNB should be Navigation Objects. That is: (NN/NR) should be not less than H_{min} , where NN is the total number of navigational Root Children, and NR is the total number of Root Children.

- Root Children text length should be less than L_{max} .
- Text NNB appearance times in the website should be not less than R_{min} .
- Root Children of a Text NNB should have similar property.
- Root Children of a Text NNB should have hyperlinks all to either outside or inside; only a small percentage of deviation D_{max} is allowed.

The constant variables above such as H_{min} , L_{max} , R_{min} and D_{max} are experimental values, which may vary in different websites according to the practical detection result.

Based on the rules above, we can easily set up the algorithm for Text NNB Finding. Small deviations are allowed for the binary conditions in practical detection. That is, even if one of the values is slightly on the wrong side of a threshold, the corresponding Text NNB can be accepted if all the other values are well away from their thresholds.

Experimental results of NNB detection can be found in Figure 11, where the NNB is removed in adapted content. For more results please refer to our website listed in Section 4.2. Experimental results show that our detection algorithm is pretty precise for Text NNB.

3.2.3 Specific FOM Generation for CO – Example 2: Page Category Detection

As described above, web page has two basic categories: Index Page and Content Page. Here we present a hyperlink based page category detection algorithm, which is effective for all languages based on XML.

We first define the Out Degree (OD) and In Degree (ID) of a web page. OD is the number of hyperlinks inside the page. ID is the number of web pages with hyperlinks to the current page in the whole website.

By statistical analysis, we find that a page with relatively large OD or ID may be an Index Page and a page with relatively small OD and ID may be a Content Page. That is, for a page with Out Degree OD and In Degree ID, the following rules exist:

If $OD > OD_0$ or $ID > ID_0$, the page is an Index Page;

If $OD < OD_0$ and $ID < ID_0$, the page is a Content Page,

where OD_0 and ID_0 are two constant variables determined by the website. To find OD_0 and ID_0 , we first sort the pages by OD and ID in descending order, respectively, and draw the corresponding $OD(i)-i$ and $ID(i)-i$ diagram of a website (i is the ordered number of a web page). Figure 4 gives an example of $OD(i)-i$ and $ID(i)-i$ diagram for Yahoo! Chinese news website (1,615 pages). By statistical analysis to many websites we find that OD_0 and ID_0 are generally the inflexions in $OD(i)-i$ and $ID(i)-i$ diagram as shown in Figure 4.

To get OD_0 and ID_0 we have the following two methods:

1) Beeline simulation. As shown in Figure 5(a), we simulate the diagram with two beelines. That is, the average distance between the beelines and the original curve of OD is minimized. Then the y-coordinate of the intersection of the two lines is OD_0 .

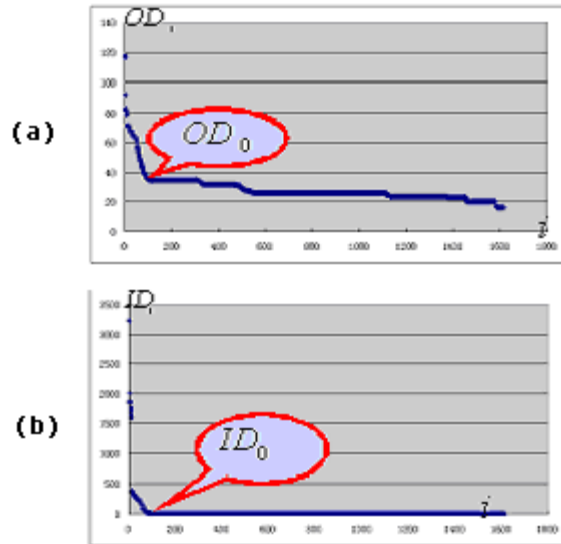


Figure 4. (a) $OD(i)-i$ of Yahoo Chinese News website
(b) $ID(i)-i$ of Yahoo Chinese News website

2) Polynomial simulation. As shown in Figure 5(b), we simulate the diagram with a polynomial curve (second power or more). That is, the average distance between the polynomial and the original curve of OD is minimized. Then the y-coordinate of the inflexion with the largest curvature of the polynomial curve is OD_0 .

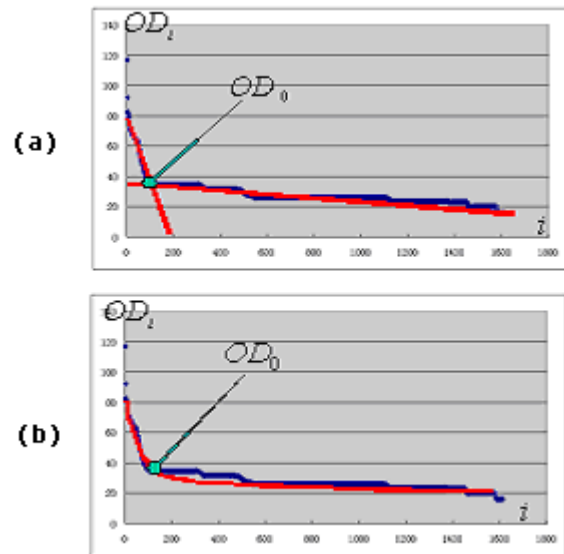


Figure 5. (a) Beeline simulation for OD_0 detection
(b) Polynomial simulation for OD_0 detection

Following is an example of our experiment results to an English news website:

Total number of web pages: 73, in which number of Index/Content Pages: 5/68

OD0 =22, ID0 =4

Detection Precision: 100%

Figure 10 and Figure 11 give examples of an Index Page (OD=36, ID=76) and a Content Page (OD=8, ID=3) in this website. Experiments to other websites have also shown promising results. For more results please refer to our website listed in Section 4.2.

Since Navigation Bars may exist in both Index Pages and Content Pages, we can take them away from web pages before calculating OD and ID of each page to eliminate the influence of Navigation Bar. With this method, we have even higher precision for page category detection. Compared to other approaches [21], our method has not only a higher precision but also good extensibility.

4. CONTENT ADAPTATION BASED ON FOM

For practical adaptation, FOM has two level guidelines. Section 4.1 presents the general rules based on Basic FOM. The specific rules for web adaptation over WAP based on Specific FOM are illustrated in Section 4.2 as an example.

4.1 General Adaptation Rules Based on Basic FOM

To provide users with the same browsing experience, we have the following basic criteria.

- Place the adapted content in the same time and space order as the original content to keep the same visual browsing experience.
- Try to align the children of a CO together to keep the content integrality.
- Keep the original layer hierarchy to make the content structure clearly.

Following are the general adaptation rules based on basic FOM.

1) Presentation Property should be considered together with device capability and language difference to generate the most suitable Media Type, Encoding Format and Layout Format for an Object.

2) Semanteme Property helps to select the most appropriate semantic layer of an Object.

3) Decoration values of different Objects are compared to decide whether an Object should be removed (Decided by Decision Engine [17]). If the value is 0, the Object should be retained. Otherwise, it can be removed if necessary.

4) Rules for Hyperlink Property:

- An Object should be retained if it has a hyperlink to other Object.

- An Object should be retained if it has a hyperlink to an unsupported application.

5) Rules for Interaction Property:

- Display: retain or substitute the Object according to actual situation.
- Submit/Query: the Object should be retained, discarded or substituted with its Brother Objects as a whole according to actual situation.

6) Rules for Clustering Property (for COs only):

- Complement: the CO can be divided into several Objects in different layers as shown in Figure 6 if separable. This rule is reversible.

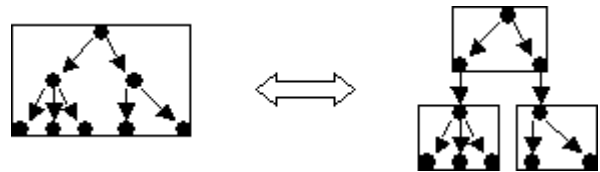


Figure 6. Rule for Complement Clustering Relationship

- Parallel: the CO can be divided into several parts at its Root Children level as shown in Figure 7 if separable. This rule is reversible.

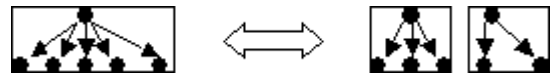


Figure 7. Rule for Parallel Clustering Relationship

- Table: The CO can be divided into several sub-tables along column (row) dimension as shown in Figure 8 (Original table is in the middle). Each sub-table contains the column (row) header and one or several rows (columns) of the table.

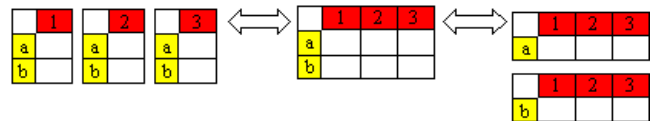


Figure 8. Rule for Table Clustering Relationship

4.2 Web Adaptation over WAP Based on Specific FOM

The rules above are just basic guidelines. They should be combined with specific rules based on Specific FOM and application environment for content adaptation. Following we present an example of adapting web content over WAP to illustrate the specific rules based on Specific FOM.

WAP environment is quite different from that of web, which requires content re-authoring and truncation to enable WAP users

to browse the web content. However, the narrow bandwidth, small memory, different protocol and poor presentation capability (small screen size, poor support for multimedia, etc.) of WAP device prohibit the delivery of normal web contents, and a web page originally designed for desktops has to be divided into several decks for WAP devices. All these make web content adaptation over WAP a significant hard problem.

Many research results have been presented in this area. W3C's "HTML 4.0 Guidelines for Mobile Access" [18] is a series of suggestions of adapted authoring for WAP. However, it has no consideration to the existing web contents. Marcin etc. [19] defined some tag-based conversion rules, but it is not a general method. Eija, etc. [20] designed an HTML/WML conversion proxy server. However, the proxy server only works for some certain applications.

Generally speaking, most of the previous approaches are Tag/Page based. They are only suitable for HTML documents and lack an overview of the whole website. Our FOM based approach provides a good solution towards these problems. Following are the specific rules for WAP adaptation based on Specific FOM.

- Navigation Bar
 - a. In Index Page: since Index Pages mainly serve as navigation guide, Navigation Bar should be retained. Text NNB can be retained without change. Image NNB, Frame, Menu or Map should be converted into Text NNB.
 - b. In Content Page: since the purpose of Content Page is to deliver semantic information, Navigation Bar can be discarded. Otherwise, the small screen of WAP device would be filled with redundant information.
- Navigation List: retain
 - a. Normal List/Info. List/Intro. List: no change
 - b. Map: convert to hyperlink text list
- Independent Navigation List: retain
- User Query/User Submit: should be retained as a whole and the server side program should be revised to adapt to WAP environment.
- Decoration Object:
 - a. Highlight: replaced with big font text
 - b. Separator/List Leader/Background: discard
- Special Objects (AD, Logo, Contact, Copyright, Reference): discard or change to text form according to authors' intention respectively

In addition to the specific adaptation rules, we have also language conversion rules (HTML to WML). Since this is not much related to FOM, here we just introduce four basic rules:

- HTML Tags with similar syntax in WML are retained without change, e.g., <p>, <u>, <i>, , , <big>, <small>, ...
- HTML Tags which can be replaced by one or several related WML Tags are converted into the corresponding WML Tags, e.g., ... to <p>...</p>, ...
- HTML Tags which are invalid in WML are discarded, e.g., <embed>, <applet>, ...

- Complex HTML Tags are converted into a set of related WML Tags according to actual situation, e.g., <table>, ...

Besides, to help users locate information quickly, the most important information should be delivered first. This can be achieved by reorganizing the web content into WML decks.

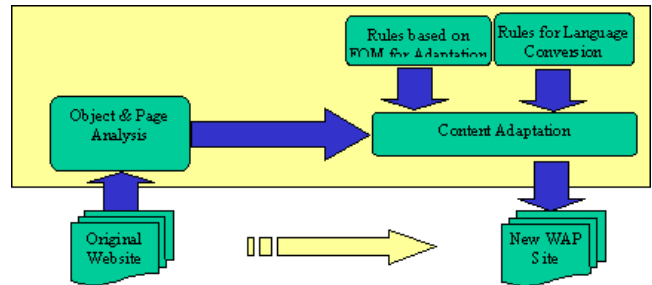


Figure 9. System architecture for WAP adaptation

The system architecture for WAP adaptation is shown as Figure 9. Object & Page Analysis module generates FOM for original website. Content Adaptation module makes adaptation according to FOM based adaptation rules and language conversion rules.

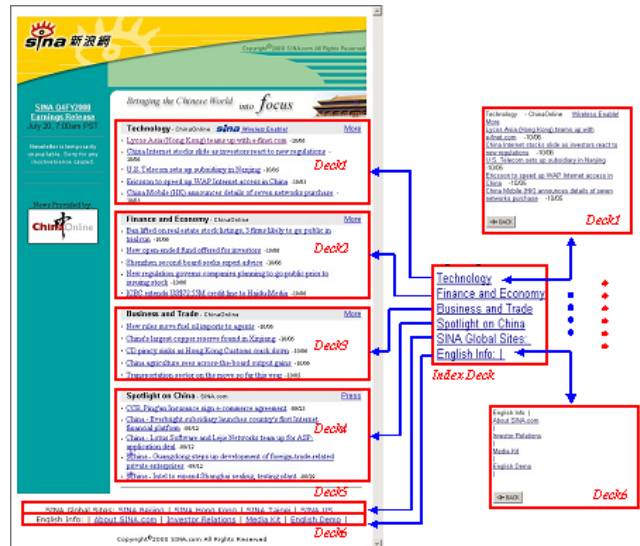


Figure 10. An Index Page and the adapted result

Figure 10 shows an adaptation example for an Index Page, in which contents with unsupported media type are removed, and several new decks are generated to show the original content at different layer. The Index Deck is an outline of the original page. Detailed contents are in Deck 1-6. To simplify the adaptation process, here we only use decks to present the adapted content for WAP device, and cards are not considered. Figure 12 shows the adaptation result in a WAP simulator. Figure 11 shows the adaptation result of a Content Page, in which redundant contents are removed, and the original page is divided into two decks because of the memory limitation of WAP device. The two

examples in Figures 10 and 11 are all practical adaptation results for an existing website. For more results please visit our website:

- <http://www.research.microsoft.com/users/jlchen/main.htm>
(Original website)
- <http://research.microsoft.com/users/jlchen/main.wml>
(Adapted WAP site for WAP phone)



Figure 11. A Content Page and the adapted result

Compared to other approaches, our approach has more satisfactory results and brings the same browsing experience to users. Since authors' intention is well understood through FOM

analysis, our adaptation is quite reasonable. For example, page function analysis (index/content page) has helped to decide whether to keep a navigation bar or not for WAP device (See Figures 10 and 11). Since the major purpose of an index page is to provide navigation guide to users, the navigation bar is retained in the index page (Figure 10). On the contrary, the major purpose of the content page is to provide information to users, and the navigation bar can be considered as redundant information, therefore, it is removed in the content page (Figure 11). Another example is the generation of the index deck (Figure 10). The system first detects the navigation lists in the index page through FOM analysis. At the same time, the parallel clustering relationship of these navigation lists are also detected. Then the system can extract the abstract information from these navigation lists and list them equally to generate the new index deck, which enables users to navigate the adapted content easily.

Besides, based on visual information and functional property analysis instead of tag analysis, our method is a general approach to various websites. It not only is able to handle HTML based websites, but also can be easily extended to other web environment. For all the contents we listed in our website, we always get relatively stable and reasonable adaptation results. Compared to our approach, previous tag-based approaches are limited to some certain applications due to the misuse of HTML tags.



Figure 12. An Index Page and adapted result in WAP simulator

5. CONCLUSIONS AND FUTURE RESEARCH

This paper presents Function-based Object Model for website adaptation. By representing Objects with their functional properties and categories, FOM effectively avoids the ineffectiveness and limitation of approaches based on semantic structure analysis. Experiment results show that our approach has more reasonable adaptation results and can be applied to various web contents and applications. As a general approach, FOM can be easily extended to the whole website and other application environment. In addition, the automatic FOM generation method also enables our approach quite effective.

Further research includes extending FOM to website function analysis, combining FOM with semantic representation model (such as XML) and user model to provide a more personalized service. Besides, we will also try to set up an automatic test bed so that users can submit the links of web pages that they are interested in and test the adapted results.

6. REFERENCES

- [1] Brusilovsky, P. Methods and Techniques of Adaptive Hypermedia. *User Modeling and User Adapted Interaction* 6, 2-3, 87-129.
- [2] Fox, A., Gribble, S.D., Chawathe, Y., and Brewer, E.A. Adapting to Network and Client Variation Using Infrastructural Proxies: Lessons and Perspectives. *IEEE Personal Communication* 5, 4, 10-19.
- [3] BickMore, T.W., and Schilit, B.N. Digestor: Device-independent access to the World Wide Web. *Proc. WWW6* (1997), 655-663.
- [4] Angin, O., Campbell, A.T., Kounavis, M.E., and Liao, R.R-F. The Mobeware Toolkit: Programmable support for adaptive mobile networking. *IEEE Personal Communication* 5, 4, 32-43.
- [5] Ma, W.Y., Bedner, I., Chang, G., Kuchinsky, A., and Zhang, H.J. A Framework for Adaptive Content Delivery in Heterogeneous Network Environments. *Proc. MMCN2000* (SPIE Vol. 3969)(San Jose, USA, 2000), 86-100
- [6] Hammer, J., Garcia-Monlina, H., Cho, J., Aranha, R., and Crespo, A. Extracting semistructured information from the web. *Proc. Workshop on Management of Semistructured Data (PODS/SIGMOD'97)*(May 1997), 18-25.
- [7] Ashish, N., and Knoblock, C. Wrapper generation for semi-structured Internet sources. *Proc. Workshop on Management of Semistructured Data (PODS/SIGMOD'97)*(May 1997), 8-15.
- [8] Simth, D., and Lopez, M. Information extraction for semi-structured documents. *Proc. Workshop on Management of Semistructured Data (PODS/SIGMOD'97)*(May 1997), 117-121.
- [9] Nestorov, S., Abiteboul, S., and Motwani, R. Inferring Structure in Semistructured Data. *Proc. Workshop on Management of Semistructured Data (PODS/SIGMOD'97)*(May 1997), 39-43.
- [10] Embley, D.W., Jiang, Y., and Ng, Y.K. Record-Boundary Discovery in Web Documents. *Proc. SIGMOD'99* (Philadelphia PA, 1999), 467-478.
- [11] Lim, S.J., and Ng, Y.K. An Automated Approach for Retrieving Hierarchical Data from HTML Table. *Proc. CIKM'99* (Kansas City, MO, 1999), 466-474.
- [12] W3C Cascading Style Sheets.
<http://www.w3.org/Style/CSS/>.
- [13] Smith, J.R., Mohan, R., and Li, C.S. Scalable Multimedia Delivery for Pervasive Computing. *Proc. ACM Multimedia 99* (1999), 131-140.
- [14] W3C Extensible Markup Language (XML).
<http://www.w3.org/XML/>.
- [15] W3C Extensible Stylesheet Language (XSL).
<http://www.w3.org/Style/XSL/>.
- [16] Easter, M., Kriegel, H-P., Sander, J., and Xu, X.W. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *Proc. ACM KDD'96* (1996)
- [17] Chen, J.L., Yang, Y.D., and Zhang, H.J. An Adaptive Web Content Delivery System. *Proc. AH2000* (Tronto, Italy, 2000) Springer Press, 284-288.
- [18] W3C HTML 4.0 Guidelines for Mobile Access.
<http://www.w3.org/TR/NOTE-html40-mobile/>.
- [19] Metter, M., and Colomb, R. WAP enabling existing HTML applications. *Proc. AUIC 2000* (2000), 49-57.
- [20] Kaasinen, E., Aaltonen, M., Kolari, J., Melakoski, S., and Laakko, T. Two Approaches to Bringing Internet Services to WAP Devices. *Proc. WWW9* (2000), 231-246.