

# An Experimental Study on Validation Problems with Existing HTML Webpages

Shan Chen

Dan Hong

Vincent Y. Shen

Department of Computer Science  
Hong Kong University of Science and Technology  
Clear Water Bay, Kowloon  
Hong Kong, China  
{chenshan, csdhong, shen}@cs.ust.hk

## Abstract

In this paper we report the results of an experimental study on the validation problem of existing HTML pages in the World Wide Web. We have found that only about 5% of webpages are “valid” according to the HTML standard. An “invalid” webpage may be rendered differently by different browsers; it may not be machine-processable; it might not be translated into other Web document formats correctly using some of the translation engines available on the Web; etc. Through sampling and analyzing the results, we have identified the most common problems in existing webpages that made them invalid. We hope our discovery can encourage standard bodies, browser developers, authoring tool developers, and webpage designers to work together so that the number of valid webpages continues to grow and the Web can indeed reach its full potential.

## Categories and Subject Descriptors

H.5.2 [Information Interfaces and Presentation]: User Interfaces - *Standardization*. I.7.5 [Document and Text Processing]: Document Capture - *Document analysis*.

## General Terms

Experimentation, Verification

## Keywords

HTML Specification, HTML Validator, HTML tidy tools, W3C, webpage characteristics, Web Sampling.

## 1. Introduction

Invented in the early 1990’s, the World Wide Web is now one of the primary media through which people publish and obtain information. As one of the most popular applications on the Internet, the number of webpages continues to grow rapidly. In 1999, there were 800 million webpages in the world [7], and according to Internet Domain Survey of the Internet Systems Consortium [6], there were already more than 233 million domains by the end of January 2004.

The importance of the Web attracted many researchers to study its properties. People are interested in the average size of an HTML document, the average number of images in a webpage, the

average size of a website, and so on [7] [8]. Our concern is about the quality of the HTML documents in today’s webpages in terms of conformance to public standards.

The HTML is actually an extensible computer language used to instruct the browser on how to render a webpage, which may contain text, images, links, and multimedia contents. To gain a competitive advantage in the “browser battle” each tool vendor tried to introduce features that would distinguish its own products. With these new features, webpage designers could easily create exciting webpages, and browser users could enjoy these pages with a brand-new experience. Unfortunately, features introduced by one particular vendor might not be recognized by the browser from another vendor. To create some order in a chaotic situation, the World Wide Web Consortium (W3C) released the HTML (Hyper-Text Markup Language) Recommendation in 1995 for resolving interoperability concerns for browsers from different vendors [12]. Using the “elements” and “attributes” defined in the HTML Recommendation, browsers parse the HTML document, analyze the tags and render them (hopefully) as the webpage designer has intended. This “Recommendation” has become the *de facto* “standard” in the Web community.

The HTML documents that follow the standard or the “valid” HTML files have a higher chance to be rendered consistently by different browsers and across different platforms than those that do not. An invalid HTML file may not be treated the same way. All the major browser vendors claim to handle valid HTML files correctly. It makes good sense for the developer of a new website to start with the standard and follow it through later revisions.

In addition to possible different appearances when rendered, invalid HTML files could be less accessible than valid ones. For example, the standard defines a required attribute “alt” in element “IMG” where “alt” provides some textual information about the image. If the browser does not support images, the value of the “alt” attribute is displayed instead. The textual information can also be read out by a screen reader to assist the vision-impaired user who cannot see the image clearly. An invalid webpage missing the “alt” attribute may not be accessible by certain users.

The Web has now become a large resource of information. More and more people use the computer to extract relevant information from the Web for their applications. An invalid HTML file often causes difficulties for the application in understanding the intended structure of the file. Information contained in such files may not be available to these applications, defeating the purpose of publicizing them on the Web. For example, the input interface of a

Web application is usually presented as a “FORM” element in the HTML file and the element “TABLE” is widely used to control the layout of the page. Consequently webpage designers often use “FORM” and “TABLE” together to provide interactivity as well as to control the layout. However, if the webpage designer does not understand the HTML language well, it is easy to commit errors while using “FORM” with “TABLE”. Such errors may prevent the interactive page to be machine-processable.

To assist webpage designers, several organizations have developed “validators” to check if a webpage conforms to the HTML standard. These validators can be used to analyze a webpage (for example, the W3C markup services [14] and the WDG validator [16]), or to analyze a whole website (for example, the CSE HTML Validator [1]). There are also a number of HTML “tidy” tools which can be used to tidy up an HTML document by correcting most of the errors in it [10]. However these tools are often not powerful enough to correct all errors; sometimes the appearance of a webpage is altered after being tidied up. Proofreading the resulting document after processing by a tidy tool could be a challenge to the webpage designers.

We wish to find out the proportion of current webpages that are valid HTML documents, and the most common problems that cause a document to become invalid. This issue was of interest to the Web Design Group; they identified six “common” problems [15] by inviting comments from other validator users. However, there was no indication how serious each problem was, and there was also no report on the methodology used by various individuals in identifying these problems. It is difficult to draw quantitative conclusions from these experiential results. To really understand how serious the validity problem on the Web is today, and to study the statistics of validation problems, we performed an experimental study by sampling the existing webpages, using a validator to check them, and analyzing the results. Our study shows that only about 5% of webpages are “valid”. We also identified the major validation problems in existing HTML webpages. We hope our findings will alert the webpage designers of the validity problem, and will help them to identify the most common errors which they should try to avoid. We hope our findings will also alert the Web application developers so that they could handle the common errors properly in order to provide better service. We also hope our findings are useful to the standard bodies and the authoring tool vendors.

The rest of the paper is organized as follows. Section 2 presents how we sampled the World Wide Web and describes the set up of our experiment. Section 3 presents experimental results, showing only about 5% of the webpages are valid. It also identifies the most common problems causing an HTML file to fail the HTML validator. Section 4 analyzes the reasons which caused these problems and provides some possible solutions. We present our conclusions in Section 5.

## 2. Experiment Set Up

Our main purpose is to study the common problems in current webpages. If we can test all the existing webpages, then the results will be accurate and convincing. Unfortunately this is not feasible since there are too many pages on the Web. If we could test random samples of this huge population, the test results would reflect the state of the existing webpages.

The unbiased selection of a random subset of the Web has been an open issue. In 1996 Bray [4] used a self-defined “start” set of

about 40,000 webpages, “crawled” the Web to measure it, and obtained some statistics on the size of the Web, its connectivity, richness, and the distribution of URL links. Bharat and Broder [3] suggested a methodology for finding a page at random from a search engine index in 1998. Their approach was based on queries to the search engine using random words. In 1999 Lawrence and Giles [7] used randomly-generated IP addresses to estimate the information available on the Web, such as the characteristics of the hosts and pages. In 2000 Henzinger, Heydon, Mitzenmacher, and Najork [5] suggested performing a random walk of the Web using multiple search engines to extract representative samples of webpages.

In this paper, we adopted the methodology of using randomly-generated IP addresses as samples due to its simplicity. We further compared our test results using this *Random IP List* with the result using the *Most Popular 10000 Websites List* made available to us by Alexa.com as additional supporting evidence. Note that both lists enabled us to analyze the properties of only the home pages of the selected websites. Although problems with the homepage may be indicative of the problems of other webpages accessible in that website, we created yet another *URL List* using search engines to confirm our findings. We repeated the test using the *Second Random IP List* in two months to see if there is any change of results over time. The details of the lists are described below.

### 2.1 Sample Lists

#### 2.1.1 Random IP List

We developed a program to randomly generate an IPv4 address. By setting up an HTTP connection to this IP address at port 80, which is the default port for HTTP connections, and listening to the response, we would know whether it was a Web server at this IP address. If it was indeed a Web server, we added it to the list.

We encountered some difficulties using this approach. Since most IP addresses were addresses for online computers which were not Web servers, most of our probes would time out. After a short while our service provider received complaints about our activities since our probes were considered hacker’s probes by some organizations. We had to stop the collection process after obtaining a list of only about 1,100 IP addresses.

#### 2.1.2 Most Popular 10000 Websites List

There are several URL lists available on the Web, such as the “500 most popular websites” from Alexa [2]. The validity of such websites is important since, if ranked correctly by the list providers, they are frequently accessed by individuals or computers. Their properties have higher impact on rendering, accessibility, and interoperability issues than a webpage selected randomly. We appreciate the support of Alexa.com who, after understanding the goals of our research project, provided us with the *Most Popular 10000 Websites List*. The results obtained using this list supported our findings using the *Random IP List*.

#### 2.1.3 URL List

Both the *Random IP List* and the *Most Popular 10000 Websites List* allowed us to check the home pages of these websites only. Since homepages constitute only a small portion of the whole Web, we wish to study the properties of other webpages to see if they are consistent with the properties of homepages.

Since most people access webpages using search engines, studying the properties of a sample of webpages indexed by some search engine will provide additional supporting data to our findings. Although the indexing method is proprietary information by the search engine owner, we could use a method similar to that used in [3] to obtain a sample. Instead of sending random words to a search engine, which may limit the responses to webpages with English contents only, we used a random string to match URL's of webpages indexed by that search engine.

According to RFC1738 [9], a URL may contain English letters, digits, and certain special characters (i.e., any of ; / ? : @ = & \$ % - \_ . + ! \* ' ( ) , % ). We generated a string of up to eight such characters randomly and sent it to a search engine which supported URL search, and included the results in our *URL list*. If a particular search request yielded more than ten hits, we included the first ten in our list. The search engines that we used are Google, Yahoo and Teoma. We assume they all independently developed their own indices. We realized that this list was certainly not a random sample of webpages but used that anyway to confirm our findings. Note that this list of 31,540 URL's did not contain our own biases.

### 2.1.4 Second Random IP List

We also repeated the random IP address generation methodology two months later and got the *Second Random IP List* over a period of one month. We used two different machines with forty threads each and managed to bypass the restrictions imposed by our service provider earlier. This list has 1,700 URL addresses.

## 2.2 The Experiment

A Validation Test program was developed in Java. Two major tools were used in this program: the W3C Markup Validator [14] and the JTidy parser [10].

The W3C Markup Validation Service is “a free service that checks documents like HTML and XHTML for conformance to W3C Recommendations and other standards” [14]. The validator uses a CGI script to fetch the given URL, parses it, and post-processes the resulting error list for easier reading. It returns a webpage which contains the error messages.

The JTidy parser is a Java port of HTML Tidy, an HTML syntax checker and pretty-printer. JTidy can be used as a tool to clean up malformed or invalid HTML documents. The version of JTidy we used is 04aug2000r7-dev.

Figure 1 shows the workflow of our experiment.

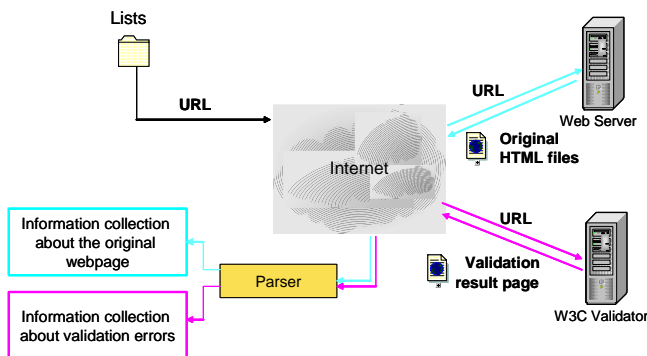


Figure 1 Workflow of our test tool

Our program sequentially picks up a URL in a sample list. The original webpage returned by the Web server is parsed by the JTidy parser to get certain information about the webpage tested: the “IMG” counter, the “FORM” counter, the W3C validation icon counter, etc. Then the URL is sent to the Validator, and the returned result page is parsed to record the error information. We stored the results in a file for further analysis.

After we got the error logs from testing the sample lists, we were able to draw an initial conclusion of the common validation errors.

## 3. Experimental Results

With the Web sample lists obtained in Section 2, we used the W3C validator<sup>1</sup> to check the errors.

### 3.1 Analysis of results from the Random IP List

#### 3.1.1 Error Definitions

From the webpages we tested, our program was able to catch about 98% of the total errors reported by the W3C Validator. Figure 2 shows the error distribution in the *Random IP List*. Note that our list includes the six “common” problems [15] identified by WDG. A more detailed analysis shows that some of these six problems are not affecting a significant number of existing webpages.

We explain some of the most common errors below. All the figures shown in this section are based on the data from the *Random IP List*.

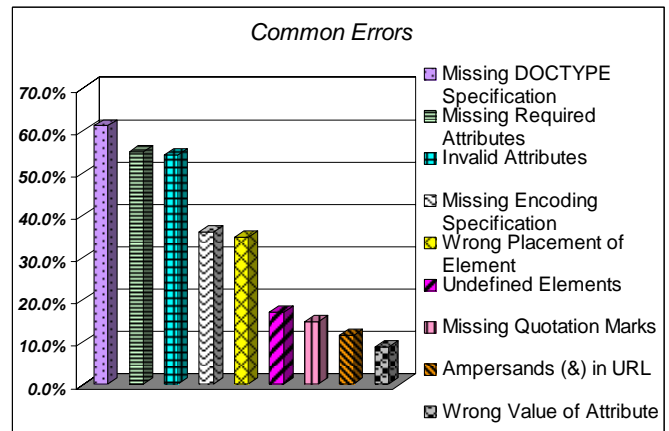


Figure 2 Most common errors

#### 3.1.1.1 Missing DOCTYPE Specification

The DOCTYPE statement is a machine-readable statement in the HTML document which specifies the structure, elements, and attributes used in that document [11]. The W3C QA Activity maintains a *List of Valid Doctypes* [13] to choose from. A valid DOCTYPE is given below:

```
<!DOCTYPE HTML PUBLIC
"-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

<sup>1</sup> Since May 2004, W3C updated their validator several times. The validators we used are V0.6.5, V0.6.6 and V0.6.7. However, these validators did not cause much difference in the results.

We found that more than 50% of the webpages omitted this important specification. Note also that the formal public identifier — the quoted string that appears after the PUBLIC keyword—is case-sensitive. An error would occur if the following is given instead:

```
<!DOCTYPE HTML PUBLIC
"../w3c//dtd html 4.0 transitional//en">
```

Since so many webpages did not have the DOCTYPE specification, the validator automatically chose the most tolerant specification (HTML 4.01 Transitional) before proceeding to parse and analyze the HTML document.

### 3.1.1.2 Missing Required Attributes

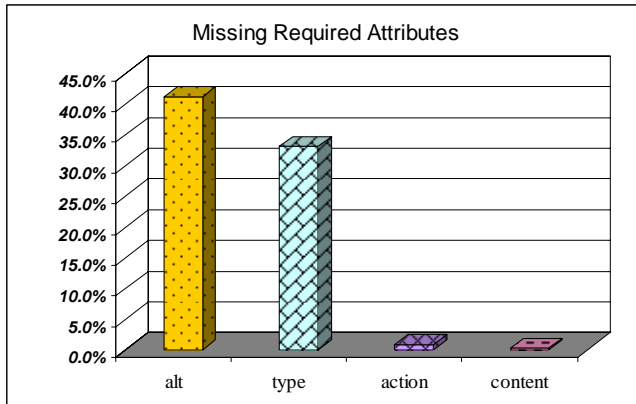


Figure 3 Missing Required Attributes errors

There are several attributes required in specific elements: for example, the attribute “alt” of “IMG” and the attribute “type” of “SCRIPT”. Since omitting these attributes causes no rendering difficulties for browsers, webpage designers usually neglect to include them. In Figure 3, we found that nearly 40% of webpages omitted the attribute “alt” while 30% of webpages omitted “type”. Even when it provides the “alt”, the value of “alt” is often an empty string. We also noticed that nearly 60% of the element “SCRIPT” omitted “type”. These two kinds of attribute errors made up 99% of the *Missing Required Attributes* errors. About 60% of all the webpages have this kind of errors.

A test on the quality of the text for the “alt” attribute is provided by <http://www.hissoftware.com/accmonitorsitetest/>. By using this validator, we find that only 38% webpages in the *Random IP List* passed this test, 17% webpages passed the test with some warnings, and the remaining 44% failed the test.

### 3.1.1.3 Invalid Attributes

In addition to omitting some required attributes, webpage designers sometimes include attributes that are not defined in the HTML standard, or defined but not for the elements they are using. Figure 4 shows the distribution of these invalid attributes. These errors can be divided into two types: those not defined in the HTML standard and those defined but not for the elements they are used.

Attributes such as “topmargin”, “leftmargin”, “marginheight”, “marginwidth” and “bordercolor” are not defined in the HTML standard. Browsers may render the webpage differently when these attributes are used in a particular element.

Attributes such as “height”, “width”, “border”, “align”, “background”, and “valign” are valid attributes in HTML. However, when these attributes are used in some elements, it may cause validation errors. For example, “background” is a valid attributes for “BODY” but <TABLE background=“test.jpg”> is invalid since “background” is not a defined attribute for “TABLE”.

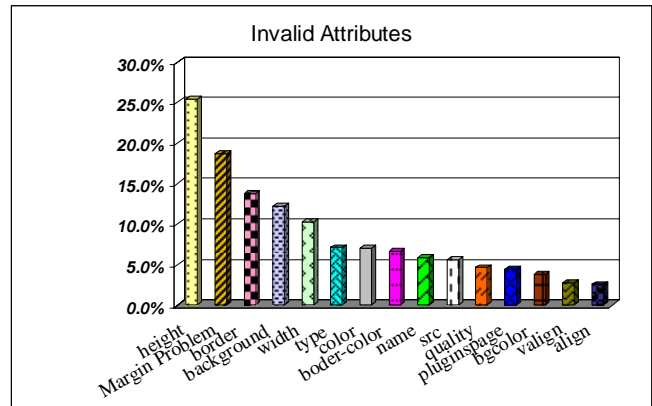


Figure 4 Invalid Attributes errors

Some of these problems are caused by certain webpage authoring tools. For example, Microsoft FrontPage, which is a widely used authoring tool, adds attributes like “topmargin”, “leftmargin”, “marginheight”, and “marginwidth” to the “BODY” element automatically when a user tries to define the page properties. This is called the “Margin Problem” by us. Such special “assistance” to the webpage designers makes the document invalid.

We found that the *Invalid Attributes* errors occur in about 60% of the webpages.

### 3.1.1.4 Missing Encoding Specification

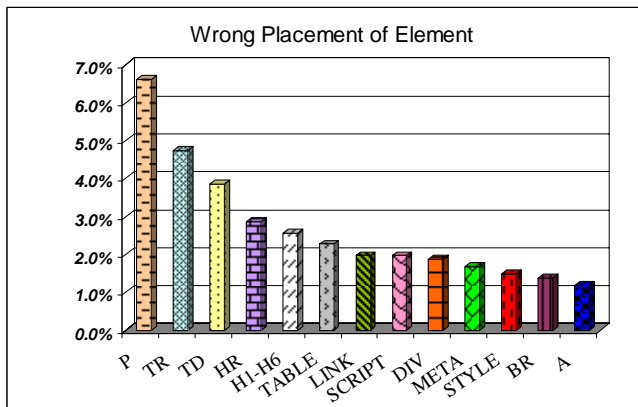
To improve interoperability, each HTML document needs to specify a character set for the document, which is called character encoding in the HTML standard. If an HTML document does not provide the encoding information (for example, a traditional Chinese character webpage), the browsers may not render the characters correctly on the webpages. The users must manually choose the encoding while browsing, which is inconvenient.

In order to validate an HTML file without encoding specification, the Validator automatically chooses UTF-8 encoding that contains most known characters. However, some characters in ISO-8859-1 (Western Europe), which is another commonly used encoding specification, are illegal or incompatible with UTF-8. If the actual encoding is ISO-8859-1 and the encoding specification is not provided in the HTML document, the Validator may fail to validate the file. Therefore, in our experiment we tried to let the Validator use ISO-8859-1 as the encoding to validate if the earlier attempt to use UTF-8 failed.

### 3.1.1.5 Wrong Placement of Element

Not only do the elements and attributes undefined in the HTML standard cause validation errors, defined elements may also cause errors if placed incorrectly. For example, it is wrong to place “META” or “LINK” element in the “BODY” section since they should be placed inside the “HEAD” section of an HTML document. Furthermore, some elements need to be placed in a

containing element. One possible cause for this kind of errors is that a block-level element (such as “P” or “TABLE”) is placed inside an inline element (such as “A”, “SPAN”, or “FONT”). A *Wrong Placement of Element* error may also occur if a previous element is not closed. About 35% of webpages have this kind of problems. The details are shown in Figure 5.



**Figure 5 Wrong Placement of Element errors**

One of the *Wrong Placement of Element* errors merits special attention: *Form with Table*. As defined in the standard, “FORM” should be specified in a container, like “TD”, instead of being specified as an element within “TABLE”. Since “TABLE” is often used to control the page layout, and “FORM” is often used for Web applications requiring user input, these two elements often are used together. At least 15% of “FORM” elements were in error when “FORM” was specified within “TABLE”, rather than in the “TD” within that “TABLE”. Unfortunately, the webpage designer might not notice the error since most browsers would render the webpage in the same look-and-feel as the designer had intended. We also noticed that nearly 10% of the “FORM” elements did not have the required “action” attribute; they were probably put there as a place holder but caused a validation error.

### 3.1.1.6 Undefined Elements

Before the HTML standard was established some browser vendors defined elements that would be useful in creating exciting webpages. One such element is “EMBED”, which could be used to insert music or video in the HTML document. Although “EMBED” has been replaced by “OBJECT” in the standard, many legacy documents still use it and most browsers still support it. Nevertheless a webpage containing “EMBED” is considered invalid by the Validator. Other frequently-used elements that are undefined in the standard include “NOBR” and “SPACER”.

### 3.1.1.7 Missing Quotation Marks

The HTML standard requires quotation marks around literal values of attributes. For example, a white background color should be specified as “bgcolor=“#FFFFFF””. Although most browsers can render the color correctly with or without quotation marks, they may render differently in other situations such as “size=“+1”” and “size=+1”. Therefore missing quotation marks could cause a webpage to become invalid. We were surprised to note that when Microsoft Internet Explorer is used to save an HTML document using the function “Save As Web Page, complete”, it removes most of the quotation marks around literal values. This browser may be the culprit of many invalid webpages.

### 3.1.1.8 Ampersands (&) in URL

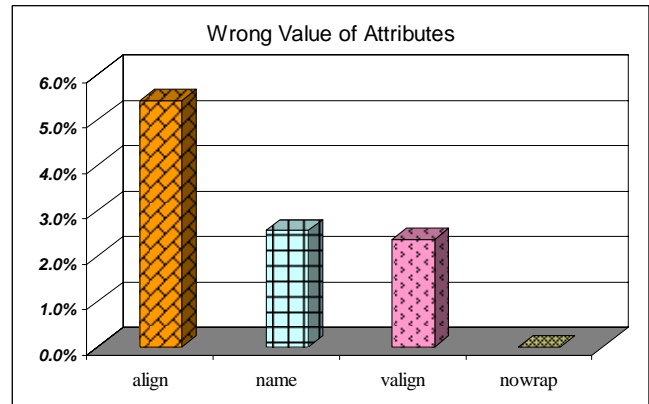
The ampersand (&) is often used in URL’s. For example,

`http://www.foo.com/user=test&pswd=test`

But if this URL is used as a hyperlink in an HTML document, it would generate an error since “&” is a reserved symbol in HTML that is the beginning of an entity. It should be changed to “&#amp;#x26;” in the hyperlink like the following:

```
<a href=
"http://www.foo.com/user=test&#x26;pswd=test"
>...</a>
```

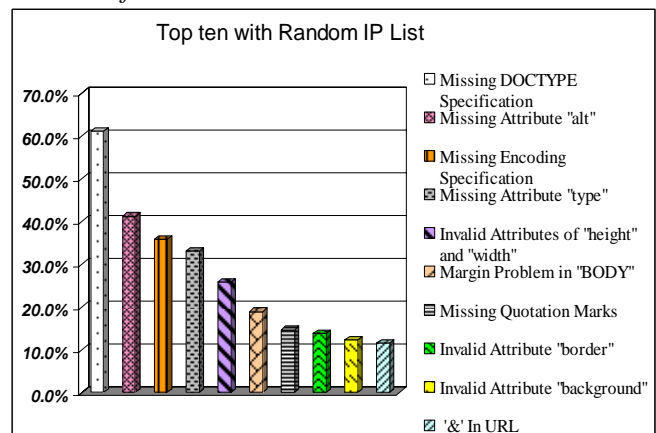
### 3.1.1.9 Wrong Value of Attributes



**Figure 6 Wrong Value of Attributes errors**

In the HTML standard some attributes have a specific range of values. For example, the value of “align” only has five choices: “bottom”, “middle”, “top”, “left”, and “right”. But webpage designers often specify its value to be “center”, making the webpage invalid. The attributes “valign” and “name” also have a similar problem. Figure 6 shows the distribution of such errors.

## 3.1.2 Top Ten Problems for HTML files in Random IP List to fail the HTML validator



**Figure 7 Top ten problems in the Random IP List**

From the analysis above, we are able to list the top ten problems that caused the HTML documents to fail the validator:

- *Missing DOCTYPE Specification*

- Missing Attribute "alt" within Element "IMG"
- Missing Encoding Specification
- Missing Attribute "type" with Element "SCRIPT" or "STYLE"
- Invalid Attributes "height" and "width"
- Margin Problem: invalid extra attributes "topmargin", "leftmargin", "marginheight", and "marginwidth" in element "BODY"
- Missing Quotation for Attribute Values
- Invalid Attributes "border"
- Invalid Attributes "background"
- Ampersands in URLs

### 3.2 Problems for the Most Popular 10000 Websites List

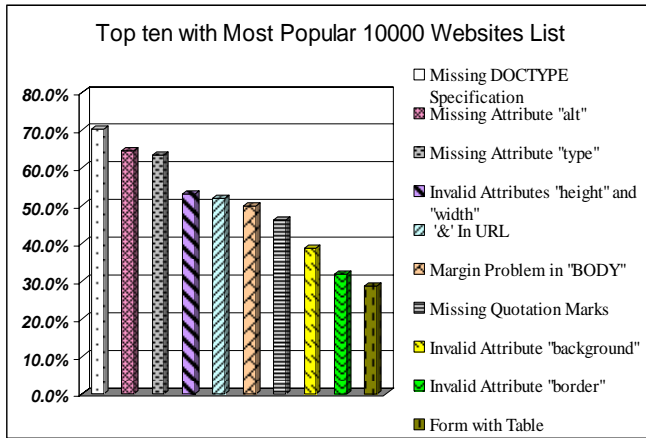


Figure 8 Top ten errors of the Most Popular 10000 Websites List

We wish to confirm our findings using the *Most Popular 10000 Websites* list from Alexa.com. The results are shown in Figure 8. Note that the top 10 problems are very similar to that from the *Random IP List*, except that the ordering has been changed somewhat. The exception is that Missing Encoding Specification has dropped out of the top ten. The Form with Table problem, which was number 33 from the *Random IP List*, enters the top 10.

We believe the webpage designers of the most popular websites are committed to make their contents readable by people around the world. Therefore they included the encoding specification so that users need not select the encoding manually.

Since many of these popular websites interact with their users, such as getting information from them, they use "FORM". They may also use "TABLE" to control the layout. Since most popular browsers support invalid usage of "FORM" and "TABLE", such errors might not have been noticed by the webpage designers of these popular websites.

Another interesting thing is that the error rate in the *Most Popular 10000 Websites* list is relative higher than the error rate in the *Random IP List*. That might be the result of frequent updates; any mistake due to an update is likely to stay unless it causes rendering difficulties.

### 3.3 Problems for the URL List

In previous tests using the *Random IP List* and the *Most Popular 10000 Websites List*, we only tested the homepages. Do all the pages have problems similar to that shown in Figure 7 and Figure 8? We used the *URL List*, which contained both the homepages and linked pages. The results are shown in Figure 9.

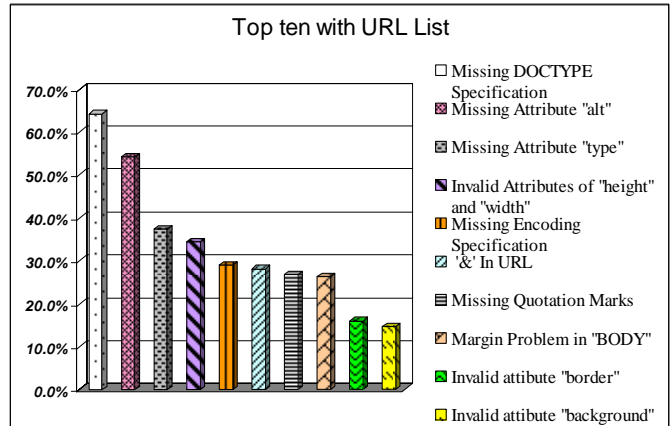


Figure 9 Top ten errors in the URL List

From Figure 9, we can find that the distribution of the top ten errors is the same as that in the *Random IP List*. The only two differences between Figure 7 and Figure 9 are the ranking of errors and the error rates.

The "Form with Table" problem is not one of the top problems compared with Figure 8. Linked pages usually provide information about some particular topics. The need for user interactions in these inner pages is much less compared to the home page of a website. That might have reduced the occurrences of such errors.

### 3.4 Further confirmation with a new Random IP List

From the analysis, we find that the top ten errors in HTML validation problem are almost the same, no matter what collection of webpages was used. We used the same method to generate the *Second Random IP List* and did the validation test three months later. The results are presented in Figure 10.

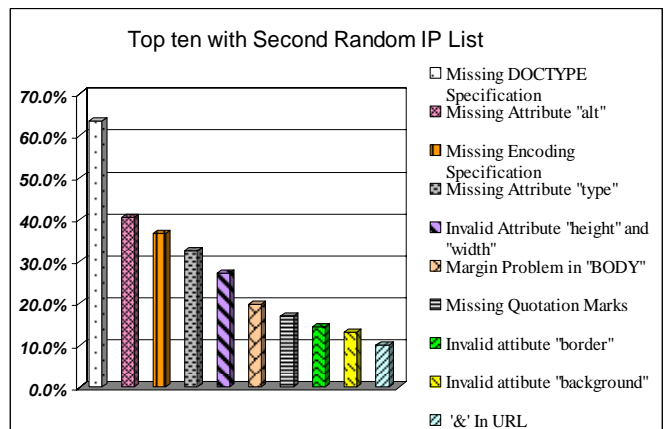


Figure 10 Top ten errors of the Second Random IP List

Comparing Figure 7 and Figure 10, we can see that the top problems in the *Second Random IP List* are the same as the top problems in the *Random IP list* and the only difference between

each other is the minor difference in error rates. After getting this data, we are more confident about the top problems that caused the HTML files to become invalid. We noticed that the top ten problems were stable although the error rate could be different. There were also minor variations in ranking.

## 4. Discussion

### 4.1 Causes of the validation problems

From the experimental results we believe there are three main reasons for an HTML file to be invalid: tolerant browsers, authoring tools that introduce non-standard features, and the culture of website development.

#### 4.1.1 Tolerant Browsers

The Web was already a vibrant media for information exchange when W3C released the HTML recommendation in 1995. Some of the existing websites had used certain features that were not later included in the standard. Although the browser developers considered the standard and would handle valid webpages correctly, as commercial product vendors they needed to satisfy the needs of their customers; i.e., to render the webpage nicely if at all possible. If a webpage designer, who may not be familiar with the standard, uses a feature that deviates from the standard, there is a good chance that the webpage will still be rendered nicely when tested with some popular “tolerant” browsers. Webpage designers are only concerned with whether browser users can get the expected rendering effect. Once this is confirmed on the popular browsers, webpage designers are reluctant to improve the code further. Such mistakes will then become part of the Web legacy. This would not have been a problem if the Web is mainly used in human-computer interactions. Tolerant browsers contribute to the problem of computer-computer interactions on the Web, which is keeping the Web from reaching its full potential.

#### 4.1.2 Zealous Authoring Tools

A great contribution of the World Wide Web is that people who are not trained in computer programming can create exciting websites which are appreciated by others. Web authoring tools (such as Macromedia Dreamweaver, Microsoft FrontPage, Netscape Composer, etc.) are of great help to those who are not computer professionals. The authoring tools assist the webpage designer to layout the website as easily as using a word processor or editor. Unfortunately some of them, possibly for reasons of aesthetics, introduce “features” that deviate from the HTML Recommendation. We mentioned the example of Microsoft FrontPage which adds invalid attributes such as “margin-height”, “marginwidth”, “leftmargin”, etc. to the element “BODY” automatically when the page property is defined. These introductions, coupled by tolerant browsers, contribute to the problem of computer-computer interactions on the Web. There seems to be a trend that the situation may be getting worse; for example, Microsoft is introducing DHTML, which is an extension of HTML that includes features on multimedia, database access, and a new object model. It will cause chaos on the Web if such features are not accepted as standard in a future version of the HTML Recommendation.

#### 4.1.3 Culture of Website Development

Since many website developers are not computer professionals, a common practice in website development is to copy the look and

feel of interesting websites and then change their contents. Mistakes in legacy websites are copied in this process. They are not identified and fixed since the tolerant browsers do not reveal them as problems. These new but invalid websites join the legacy of the Web and further propagate the problems.

### 4.2 Possible solutions

Although a number of the problems we identified, such as the *Margin Problem*, are cosmetic in nature, there are problems that delay the Web in reaching its full potential. For example, the missing “alt” attribute will make the website less accessible. Some problems, such as *Wrong Placement of Element*, may cause rendering errors or even failure in parsing the HTML document. This will affect the processing of the document by computers.

The Web is a most popular application since it brings the benefit of computers to everyone who has access to the Internet. We cannot expect the user community to solve the problem, since many webpage designers are not familiar with the HTML standard and, even if they do, they may not know how to fix the mistakes. Due to the huge legacy of existing webpages we cannot expect the browser vendors to reveal problems since if some browser cannot handle some popular non-standard features, people would not use them.

We hope our findings are useful to the standard bodies and authoring tool vendors. They should review the existing practices and reach consensus on which features should become part of the standard, and which features should no longer be supported by authoring tools. The authoring tools could do better by including the feature of some Tidy tools before the work is saved. At least they should not provide hints that would suggest the use of invalid features while the page is being created. If all authoring tools can support that the validity check before saving a document, not only new webpages will be valid but also some existing webpages, when updated or copied using the authoring tool, will be valid also. It is only through the better cooperation of standard bodies and authoring tool vendors can the Web reach its full potential.

We also noticed that an interactive process between the tidy tool and webpage designers would be better than the existing approach. For example, the tidy tool inserts “alt= ” when an image misses the attribute “alt”. This will not help users to learn more about the image if it could not be displayed or viewed by vision-impaired people. If the tidy tool can remind the webpage designer to write a brief introduction about the image, this problem can be fixed.

## 5. Conclusion

The validation problems in existing webpages have drawn more and more attention because of the increasing trend of Web communications moving from a human-to-computer process to a computer-to-computer process. In this paper, we conducted an experiment that identified the major validation problems in current webpages. We use the *Random IP List* as a basic test sample and identify the top ten problems which cause the webpages to fail the validator. We further use the *Most Popular 10000 Websites List*, the *URL list* and the *Second Random IP List* to confirm our findings. It is disappointing to find that only about 5% webpages in our sample lists are valid according to our test results. Some problems, such as the missing “alt” and the “FORM with TABLE” problems not only affect Web accessibility but also affect the move to computer-to-computer communications on the Web. We

believe our findings are useful to the standard bodies and authoring tool vendors, as well as to the Web application developers and webpage designers. Due to the diversity of validation problems, it is not going to be easy to fix all the errors in existing webpages. But we can improve the situation by removing the causes of the most common problems as suggested in our paper. Although the webpages may still be invalid after fixing the most common errors, we believe that through a deliberate effort and with the collaboration of webpage designers, standard bodies, and vendors, the validation problems could be gradually reduced, and the Web can eventually reach its full potential.

## 6. Acknowledgement

This project was supported in part by the Sino Software Research Institute grant No. SSRI 01/02.EG14, World Wide Web Consortium Office for China.

## 7. References

- [1] AI Internet Solutions, CSE HTML Validator. <http://www.htmlvalidator.com/>.
- [2] Alexa.com, 500 Most Popular Web Sites. [http://www.alexa.com/site/ds/top\\_500?p=DestTrLearn\\_W\\_g\\_40\\_T1](http://www.alexa.com/site/ds/top_500?p=DestTrLearn_W_g_40_T1), May, 2004.
- [3] Bharat, K., and Broder, A., A technique for measuring the relative size and overlap of public web search engines. Proceedings of the 7<sup>th</sup> World Wide Web Conference (Brisbane Australia, May 1998), 379-388.
- [4] Bray, T., Measuring the web. Proceedings of the 5<sup>th</sup> World Wide Web Conference (Paris France, May 1996).
- [5] Henzinger, M., Heydon, A., Mitzenmacher, M., and Najork, M., On near uniform url sampling. Proceedings of the 9<sup>th</sup> World Wide Web Conference (Amsterdam The Netherlands, May 2000), 295-308.
- [6] Internet Systems Consortium. ISC Internet Domain Survey. <http://www.isc.org/ops/ds/>, Dec 2003.
- [7] Lawrence, Steve, and Giles, C. Lee, Accessibility of information on the web. Nature 400 (July 1999), 107-109.
- [8] Lee, D.C., and Midkiff, S.F. A sample statistical characterization of the world-wide web. Proceedings of Southeastcon '97. 'Engineering new New Century' (April 1997), 174-178.
- [9] Lee, T. Berners, Masinter, L., and McCahill, M., RFC1738. <http://www.ietf.org/rfc/rfc1738.txt>, December 1994.
- [10] SourceForge, JTidy Project. <http://jtidy.sourceforge.net/>, May 2004.
- [11] W3C, Don't forget to add a doctype. <http://www.w3.org/QA/Tips/Doctype>, August 2002.
- [12] W3C, Hypertext makeup language activity statement. <http://www.w3.org/MarkUp/Activity>, January 2003.
- [13] W3C, List of Valid Doctypes. <http://www.w3.org/QA/2002/04/valid-dtd-list.html>, Dec 2003.
- [14] W3C, W3C Makeup Validation Service. <http://validator.w3.org/>, May 2004.
- [15] Web Design Group, Common HTML Validation Problem. <http://htmlhelp.com/tools/validator/problems.html>, May 2004.
- [16] Web Design Group, WDG validator. <http://www.htmlhelp.com/tools/validator/>