Characterizing Web Content

W. A. Vanhonacker[1]
Centre for Advanced Internet Architectures. Technical Report 040227A
Swinburne University of Technology
Melbourne, Australia

***Abstract*-Characterizing Web content is important for modeling. Web behavior which is in turn more crucial to the appropriate evolution of Web protocols and systems. This report gathers the results and conclusions from the analysis of a wide random set of web pages. There are several option for the focus of web characterization. Here we focus on the content; such as the structure of a web page, the size, the cachability, the number and the type of objects. We want to simplify the diverse web contents into one standard web page.**

Keywords- Web content distribution, web crawler, cachability.

I. Introduction

Part of the Inverted Capacity Extended Engineering Experiment (ICE³)project conducted at the Centre for Advanced Internet Architectures (CAIA) has been the development of tools to simulate, test and characterize inverted capacity networks. In such a network, the actual low-bandwidth last-mile becomes a high bandwidth service in the order of multi-megabits or even gigabits per seconds. The current ratio of edge to core bandwidth will be inverted. This highly increased access network allows any customer with sufficient storage capacity to act as a possible content cache for other nearby neighborhoods. Being able to push the content close to the user obviously brings a relevant decrease in download time. The goal of the research is to examine the benefits and draw backs of such an architecture, and evaluate the end result on the all important metric of web page download time.

Part of this project is to create a tool that will simulate inverted capacity and traditional networks. However, before such a tool could be developed, the answer to the question: "What constitutes a typical web page?" had to be answered. We needed to know the distribution of type, size and cachability of objects contained in average web pages so as to decide if the benefits of using an inverted capacity architecture would be worthwhile.

We found that the complex web content seen on the Internet today cannot be generalized into a single mold. Instead, different categories are needed in order to group web pages of similar genres. By finding the typical characteristics for each category of web page, we will gain insight that can be leveraged to develop the required simulation tool.

II. Background

There are a few projects that are investigating similar areas covered by the ICE³ project. However, there has been no known work done do date that attempts to characterize web pages based on the properties of the web objects they contain.

Recent projects have investigated one of the most fundamental aspects of web caching: the modeling of content modification dynamics of web objects [1][2].

Other research has investigated the content of the World Wide Web, but focused more on network layer issues such as domain names, protocol and port usage, etc. [3].

III. Data Gathering

A. The web crawler

We used a modified version of the LARM web crawler [4], which is part of the Apache Jakarta Project [5], to gather the statistics for this report. This tool crawls a user configured set of web pages and creates an index file containing the links between the URLs.

In order to collect object statistics for the visited sites, the crawler was modified to store the details of each web page into a database. The parameters the crawler records into the database are configurable. Typical parameters include number of objects, the type of the objects, the size of the objects, the cachability of the document and its objects, etc..

IV. Results

The crawler was set to follow a list of 10000 web pages and record the type, size and cachability of each object contained in each of the pages. The list of web pages to crawl was randomly selected from the logs of an Internet web proxy [6].

Note that because our crawler was going through a proxy that blocks some pages, the total number of pages stored in our database for analysis came to 8202.

We define the following terms for the context of this paper:

- a document: this is a HTML document that will be analyzed by the crawler (also called web page).
- An object: A document contains many objects (also called children), these objects can be images (jpegs or gifs), frames, scripts etc.. The document itself is also considered to be an object.
- An object type: The type of the object refers to the type of tag with which the object is linked to the document. Table 1 gives a list of the typical link

1 This work was performed while working for Swinburne University of Technology. Wendy Vanhonacker can be contacted at wendy@vanhonacker.ch

types we found with their corresponding tag. Each type are referenced by a number which was produced by the crawler we used.

Type number	Object type	Tag
3	Frames	FRAME
5	Images	IMG
6	Scripts	SCRIPT
7	Links	LINK
8	Embedded objects	EMBED
Other	Areas, Unknown	AREA, A, ?

Table 1. Object types

Our crawler requests each web page one by one, and has been configured to not follow any of the links contained in the specified web pages.

The basic output of the crawler is a table with one record for each object. Each record consists of the following fields:

- object ID: the crawler uses a hash function to generate a unique ID.
- URL: the URL of the object
- referrer: the URL of the HTML document the object comes from. If the object is the main HTML text, the referrer is null.
- Type: the type of the object is the type of link it is attached to the main document by, for example image (IMG tag) or a script (SCRIPT tag), etc...

Note that since we do not follow links between web pages, any object that is such a link is not indexed by the crawler. The net result is we are only analyzing the objects required to display a web page.

A. Document cachability

Of 8202 pages, about 5460 pages, or 66.6%, were non cachable.

B. Object type distribution

The total number of objects was 90833.

The concern here was to define how many objects a document had on average and what type they were. Figure 1 shows the frequency distribution of the typical object types within the examined documents.

Figure 2 to Figure 5 show the distribution of each objects types: images, scripts, frames, embedded objects, links.

As you can see, on those figures, most of the documents do not have a lot of objects. From further research, we found that 19% of the documents do not have any objects at all. The average number of objects is close to 10 objects, but the median is 4 objects. That is because 64% of pages had less than 10 objects, but a few pages have a high number of objects which skew my results. The maximum number of objects found is 224.

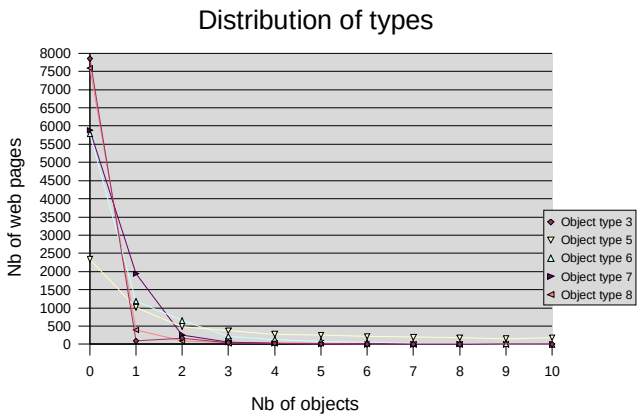

Figure 1 Distrbution of types on a total of 8202 documents

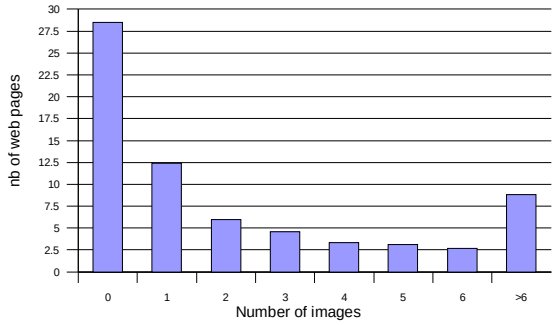

Figure 2 Frequency distribution of images (object type 5)

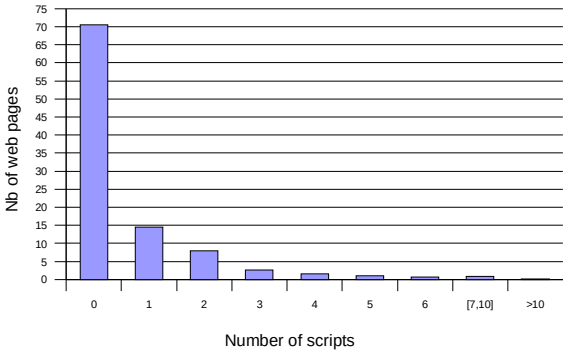

Figure 3 Frequency distribution of scripts (object type 6)

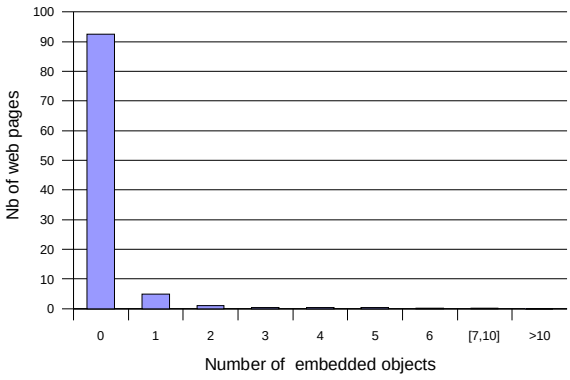

Figure 4 Frequency distribution of embedded objects (object type 8)

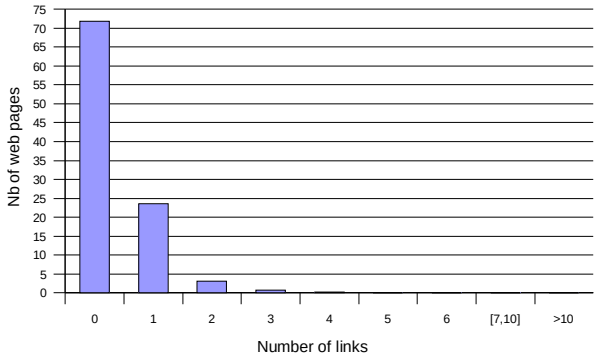

Figure 5 Frequency distribution of links (object type 7)

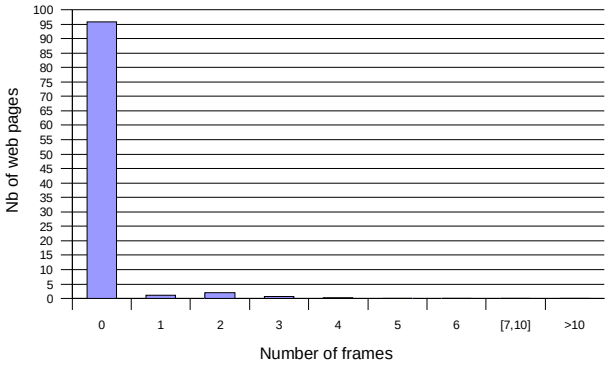

Figure 6 Frequency distribution of frames(object type 3)

C. *The size distribution*

Size of objects is possibly the most important object attribute, as it is the main influence on object download time. Figure 7 plots the distribution of the object size and separated the objects per object type.

Most objects are between 100bytes and 10kbytes in size. The average size of frames is approximately 7kbytes and the average for images, scripts and links is approximately 4kbytes.

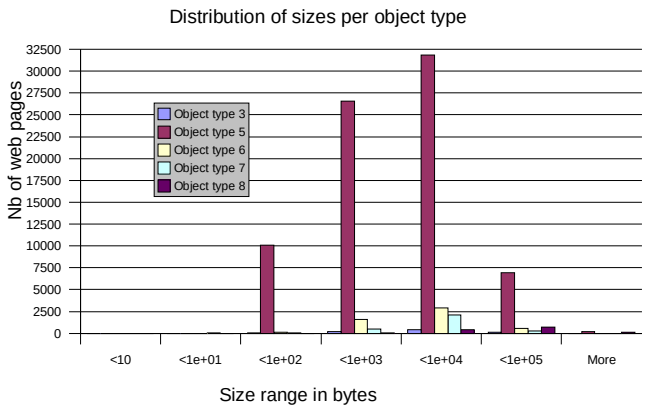

Figure 7 Distribution of size in bytes

D. *Object cachability distribution*

This part is also influential in the computation of download time. If an object is cachable, then the client can download it from a closer server (cache) instead of the remote server. The download time is thus shorter.

Figure 8 shows the distribution of cachable objects over the whole set of documents.

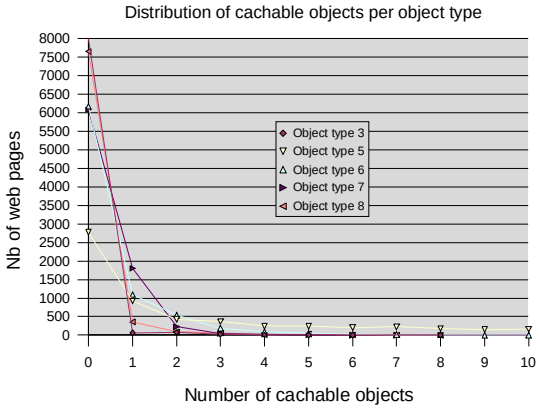

Figure 8 Distribution of cachable objects per object type

In order to have an actual feeling for these results, Figure 9 is the plot of the distribution of cachable objects versus all objects. As you can see the number of cachable objects versus the total number of objects are almost the same: We found that around 85% of objects are cachable.

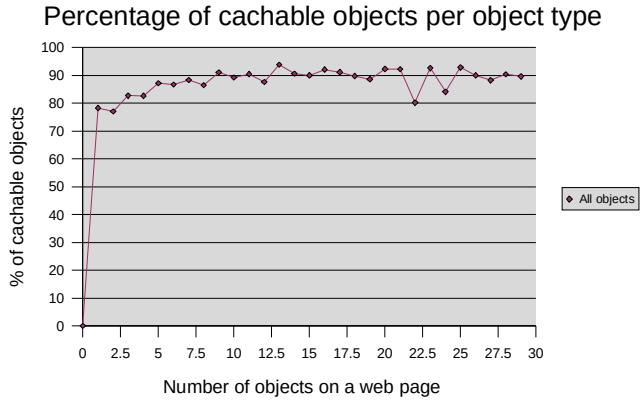

Figure 9 Percentage of cachable objects versus the number of objects

Table 2 summarizes the results. Here we compute the percentage of cachable objects overall objects of the same type. As you can see the percentage of cachable objects depends a lot on the type of the object. This is a little obvious since for example, objects of link type 3 which are frames, have a low percentage of cachability (45%): this is due to the fact that most of the time frames are generated dynamically from a query, thus not cachable.

LinkType	Total	Total Cachable	% Cachable	Total size (bytes)	Total size Cachable (bytes)	% of bytes Cachable
3	798	363	45.49	5603360	1931444	34.47
5	75773	69419	91.61	311047600	293237809	94.27
6	5220	3873	74.2	24392656	19708856	80.8
7	2933	2636	89.87	12745938	11148572	87.47
8	1208	1080	89.4	56716600	53933701	95.09

Table 2: Distribution of cachability

The percentage of cachable objects versus the percentage of cachable bytes stays pretty close together. If the percentage of cachable bytes was higher than the percentage of objects cachable, that would have meant that bigger objects are tend more to be cached than others. Since we don;t have this difference, this means that people do not really tend to cache big objects, which they should do!

E. Analysis

We could try to define a standard page from these results by taking the averages for each of the observed parameters. However, it is clear that the distribution is too diverse to generalize this much. For example, 66.6% of documents are non-cachable. If we define our standard document to be non-cachable, we automatically omit 30% of sample documents. Intuition also tells us that there are too may different types of pages on the web to fit to a single template.

To remedy this problem, we decided to separate the web documents into a few classes depending on the following variables:

- whether or not the web page is cachable
- whether or not it has children
- whether the children are all cachable, non cachable or a mixture of both.

We sorted our sample of web pages into the resulting 8 classes. Figure 10 shows the frequency distribution for the web pages belonging to each class.

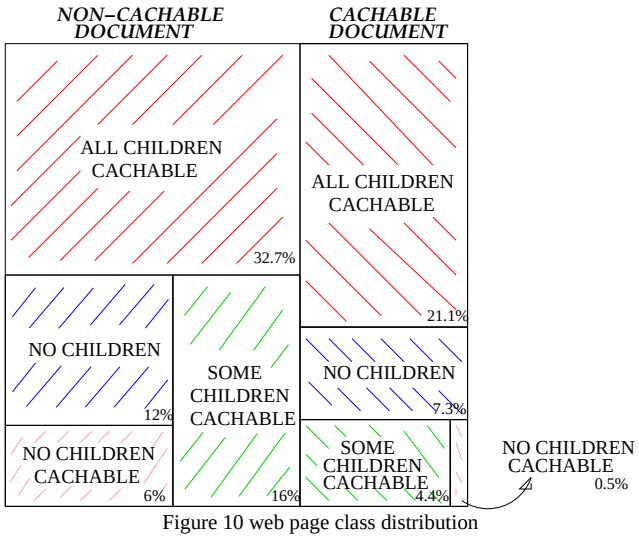

Figure 10 web page class distribution

We can thus create one generalized web page for each class; and the resulting 8 templates can then be used as a good approximation for the various types of available web content.

V. Conclusion

This report outlines the results of the analysis of a set of random web pages. The goal being the definition of a single standard web page.

In order to do so, we characterized the main concepts of a web page, such as the distribution of objects in a document, the cachability of the document and also the size and cachability of the objects.

It is possible to create a standard page by averaging every result, but we concluded that it was more relevant to have a number of classes to represent all web content.

We proposed 8 different classes, depending on document cachability, number of objects, and the cachability of the document's objects. If we create one template for each of these categories, we end up with a set of documents to approximate all web content.

Acknowledgments

Thanks go to Clancy Malcolm for his ideas and help in gathering the results and Lawrence Stewart for his help in writing this report.

References

[1] C.Chi, H.Wang, "A Generalized Model for Characterizing Content Modification Dynamics of Web Objects", September 2003, http://2003.iwcw.org/papers/chi2.pdf

[2] C.Malcolm, G.Armitage, "Dynamics and Cachability of Web Sites: Implications for Inverted Capacity Netowkrs", Center for Advanced Internet Architecture, Swinburne University of Technology,April 2003, http://caia.swin.edu.au/reports/030405B/CAIA-TR-030405B.pdf

[3] A.Woodruff, P.Aoki, E.Brewer, P.Gauthier, L.Rowe, "An Investigation of Documents from the World Wide Web",Computer Science Division University of California at Berkeley,May 1996, http://www.geckil.com/~harvest/www5/papers/P7/Overview.html#results

[4] Clemens Marscher , The LARM Web Crawler – Technical Overview", as of February 2004, http://jakarta.apache.org/lucene/docs/lucene-sandbox/larm/overview.html

[5] The Jakarta Apache Project, as of February 2003, http://jakarta.apache.org/

[6] The IRCache project, as of February 2003, http://www.ircache.net/