Analysis of Behavior of Parallel Web Browsing: a Case Study

Salman S Khan

Department of Computer Engineering
Rajiv Gandhi Institute of Technology, Mumbai, Maharashtra, India

Ayush Khemka

Department of Computer Engineering
Rajiv Gandhi Institute of Technology, Mumbai, Maharashtra, India

Sharmila Gaikwad

Assistant Professor, Department of Computer Engineering
Rajiv Gandhi Institute of Technology, Mumbai, Maharashtra, India

Abstract — When users access web-based systems, they often take advantage of what current web browsers offer: multiple tabs with different pages. Whether they are on technical concept or social concept they can browse the whole concept or concurrently browse other relevant objects in the system on Web. This behavior goes unnoticed or it is only estimated in the traditional web usage mining, where it is supposed that the user opens only one page at a time. However, if we could record the parallel browsing behavior, we could better understand the user's activity, goals and improve for personalization. The time utilization is achieved using technologies which have access to two simultaneous pages at a time. Here the ability of human tendency to do multiple tasks at a time comes into picture. The ability to do concurrent task having access to two pages on a single application window. In this paper, we introduce a model behavior for parallel browsing.
Keywords--Parallel browsing behaviour, Displaying two pages at a time, Open multiple resources in parallel

I. INTRODUCTION

A tool for travelling the internet known is as web browser. A "client software" for viewing the World Wide Web. A multimedia interface integrating of text, graphics, audio and video. A web browser acts as an interface between the user and web server. The necessity part of using the internet is the browser. With the help of browser we are able to interact with the internet. It provides certain facilities which are required for accessing the internet. Browser is the only medium to connect to internet. Without browser we are unable to access the internet when internet began the browser was also born. So a must need for internet is the browser without it the internet is unable to access. Since browser is the only medium for accessing the internet, it should provide certain facilities so that the user is able to access without any difficulties. It should have a method of user understanding so as to perform certain function. Current systems are often realized as web based systems, allowing their users to work from a range of platforms and even remotely from their home. The learners often take advantage of current web browsers and open multiple resources in parallel, weather it in multiple browser tabs or multiple browser windows.

In this way, l can open multiple objects (web pages) from the medical or through any internet domain system and switch between them. If we could record and analyze this parallel browsing, we could develop an application that would run two pages parallel. In a medical system where a researcher, a professor or any student studies or researches about any project many complicated things come into picture. They start to experience difficult things .often time they need to refer two or more things which are interconnected to each other. So they need to refer previous or current or the next possible content which is required. Similarly, when the user is learning a new topic and she or he starts with an explanation object, she or he may switch to relevant exercises and notes to practice the presented information. Relevant notes or explanations in these examples can be selected manually by a teacher during course authoring or even automatic methods can be applied, such as content-based methods using similarities between learning objects. So an application is required to match these manually done things in software.

However, knowing users' browsing behavior allows us to better model the user, what she reads in what situations and the better user model alone can be used for better information personalization for the user. Users can also find learning objects and external sources relevant to their current activity outside of the relevance expressed in the domain model of the content and outside of relations discovered by content-based methods or created by the teacher. The relevant explanations and external sources can be recommended to other users struggling on the same experiment on which the user opened them. relevant notes can be recommended to users reading an explanation; the domain model can be augmented with discovered relevance, etc. on the client side, we can exactly track what a user does by augmenting the browsing software, but this is limited only to users in a closed experiment in a lab where the software is provided by the researchers, or to a portion of users that are willing to install tracking extensions. The necessity of additional software limits its use in real settings and it does not provide information on all users. We proposed a model for parallel browsing behavior based on events observed by client-side scripting. As such, it does not need installation of any additional software (e.g. browser extensions).

II. STUDYING OF PARALLEL WEB BROWSING

Opening multiple windows and having multiple web pages open at the same time have been supported by web browsers for a long time since their soon versions. Afterwards, tabbing interfaces were added, which allowed multiple pages to be opened in the same window in a user interface elements called tabs. First, they were supported through browser extensions and later natively by browsers themselves. Since the release of Internet Explorer 7 in the 2006, all widely used browsers support tabbing. Both tabbed and multi windowed browsing methods are very similar, the main difference is that with multiple windows a user switches between opened web pages via main taskbar of the operating system and with multiple tabs, the switches are made via the list of tabs inside the browser. While differences between tabbing and multi windowing are important from the perspective of the human- computer interaction and user interfaces, in the web usage mining, we only need to know which resources are being browsed and how the user travels through them. For simplicity, we address the use of multiple browser tabs and multiple browser windows as tabbing and call both the browser windows and tabs as tabs.

Parallel browsing has been previously studied through various methods selected for given research goals. For example, in order to just survey how users work with web browsers and search engines while searching for information, a questionnaire can be used. In having multiple tabs open while searching was found to be common. There are several approaches for the tracking user behavior. Basically, we can realize user behavior tracking either as server-side, or client-side.
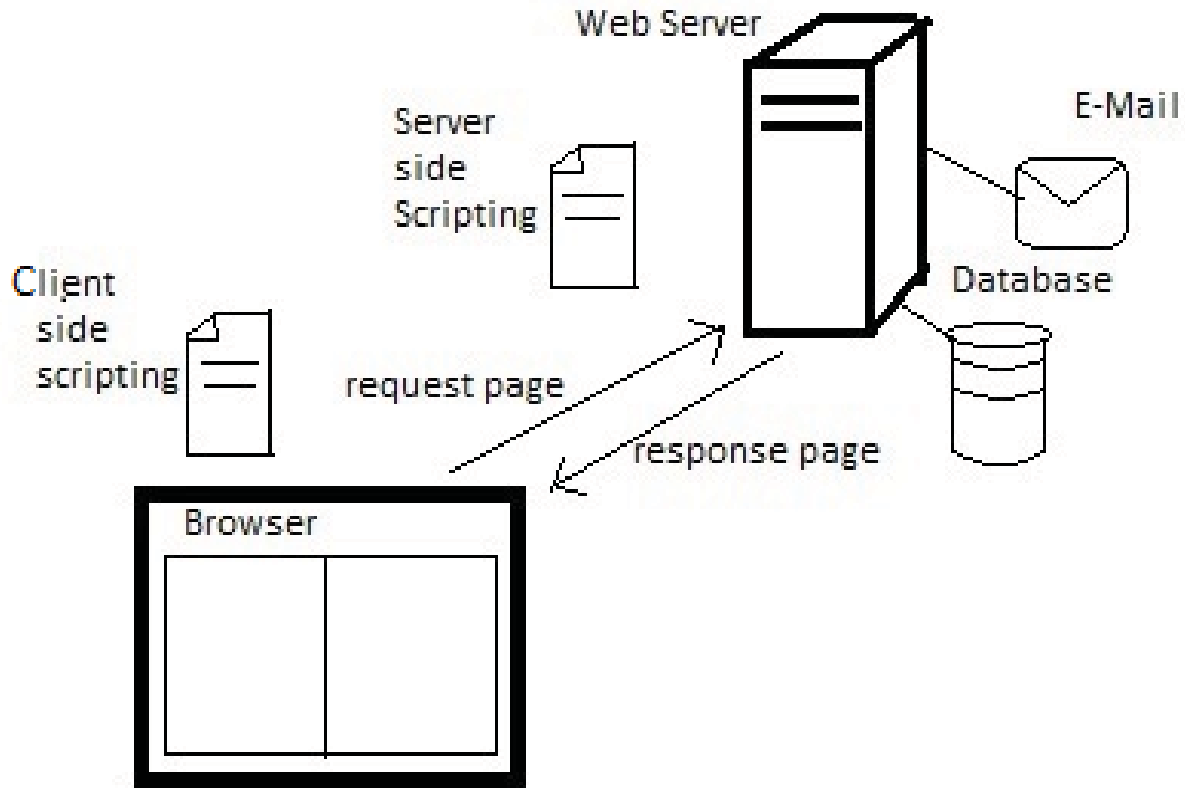
Fig 1: Working of a web browser

This figure explains the working of the web browser. The request has been send by the user to the server and the response is given back to the client. The scripting is done on both side that is on client side and server side. Server gives the necessity content from respected database. Both the side of the web browser can access independently without relying. Requested content will be displayed in both the display frame. In [1] having multiple tabs open while searching was found to be common. There are several approaches for the user behavior tracking.

A.   Server Side Tracking

In server-side tracking, server logs contain only when page loads and no direct or indirect information on tabbing. Parallel browsing can be only estimated when using this kind of tracking. In [2], authors has introduced a model for both linear browsing (action of following a link) and parallel browsing (actions of opening a tab and switching between open tabs) and subsequently used the server-side data for modeling the user behavior. If we consider linear browsing it is accurately observed from server data and only an estimate of possible click trees and paths through them can be made for parallel browsing, giving only information on how the users could be browsing in parallel, giving many possibilities. In [3], a model based on stochastic process was applied to logs of visited pages and their referrers. An improvement has been seen, but the model cannot distinguish between branching (opening multiple tabs from single page) and backtracking (opening a page in the same tab, returning back and opening another page).

B. Client Side Tracking

In [4], a browser plug-in was used with 10 participants to analyze exploratory tasks. During a search session, users have opened 2 additional tabs. Large logs from a plug-in were used in [5] (60 billion page views from 50 million users) and authors looked for situations where a user opens a page and the tab identifier changes.

In *client-side scripting*, a JavaScript code is embedded within a page and it can see how the user works with the page (where she clicks) or when she leaves. This approach collects information from every user, but the code is always running within a sandbox of the single page, in which it was embedded. This page is present in a single tab and as such, the tracking code sees no information about other tabs. In a recent study [6], tracking code was embedded in the results page of a search engine. Presence or absence and the order of page loads and clicks on the results were used. For example, a page load and two following clicks mean that the result must have been opened in another tab (branching), because the source page still existed for the second click to occur. This approach is limited to detection of the initiation of parallel browsing within the results (e.g. no data is available on switching between the result pages) and there are branching situations which are not covered by this approach.

Overall, it has not been sufficiently studied how to use the available events to reconstruct the parallel browsing with the client-side scripting methods, thus covering natural behavior of all users in their own settings in an uncontrolled experiments, where they are typically not presented with augmented software. Several studies have shown that it is important to consider, explore and exploit the parallel browsing behavior. Parallel browsing is now more common than the linear browsing. In [7], user can create 4 to 20 tabs that to with a 100 of navigation. Tracking tab usage is also important when considering traditional web usage metrics, e.g. the re-visitation. In [7], 30% of created tabs were selected multiple times, in large contrast to traditional view based on linear browsing where each loaded page is viewed exactly once. A loaded page can be never viewed (a false visit in the traditional approach), or it can be left and visited again by switching out of its tab and back to it again (a revisit invisible to traditional approach). Study in [8] reports traditionally computed re-visitation rate 39.3% and effective re-visitation with tabs actually 59.6%.

III. UNDERSTANDING THE WORKING OF THE BROWSER

The main function of a browser is to present the web resource you choose, by requesting it from the server and displaying it in the browser window. The resource is usually an HTML document, but may also be a PDF, image, or some other type of content. The location of the resource is specified by the user using a URI (Uniform Resource Identifier).

The way the browser interprets and displays HTML files is specified in the HTML and CSS specifications. These specifications are maintained by the W3C (World Wide Web Consortium) organization, which is the standards organization for the web. For years browsers conformed to only a part of the specifications and developed their own extensions. That caused serious compatibility issues for web authors. Today most of the browsers more or less conform to the specifications.

The responsibility of the rendering engine is well... Rendering, that is display of the requested contents on the browser screen. Different browsers use different rendering engines: Internet Explorer uses Trident, Firefox uses Gecko, Safari uses WebKit. Chrome and Opera (from version 15) use Blink, a fork of WebKit.
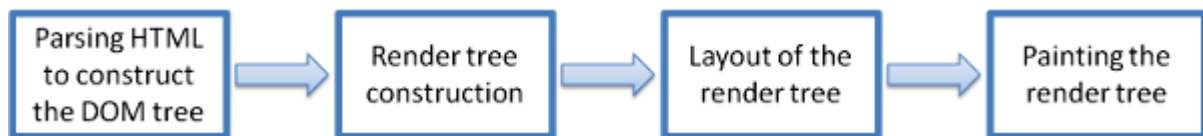

Fig 2: Flow of the Rendering Engine

IV. IMPLEMENTATION

Now the framework for displaying pages is ready, it will evaluate the loaded pages and display in browser window. The browser is divided into 2 so that two pages can be shown. One side will have one browsing window

and another side will have another browsing window. Both will have their own URL address bar to display respective pages at their side. This will make an independent source of viewing to each side.

Adding 1st & 2nd panel

sp=new JSplitPane(JSplitPane.HORIZONTAL_SPLIT,true,p1,p2);

sp.setDividerSize(10);

sp.setOneTouchExpandable(true);

sp.setDividerLocation(625);

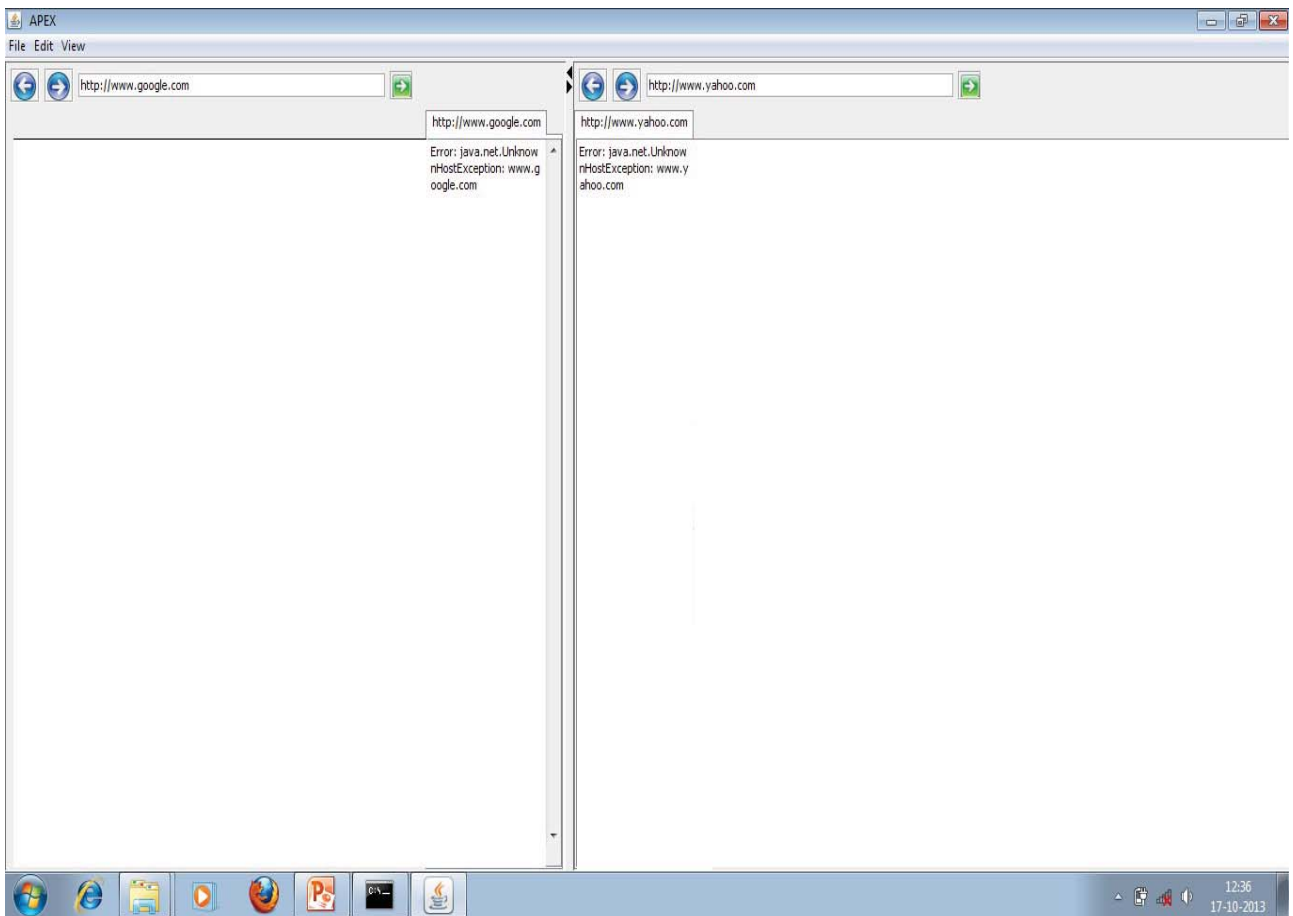*con.add(sp,BorderLayout.CENTER);*

Fig 3: Overlook of the browser

As shown in the figure it describes an interactive way browsing through web browser. One side is displaying one page another side is displaying another page. Which help the user as if for the medical person who requires two or more reference. As there are many link connected to each in medical term this will help to achieve those links. They don't have to hide one page and stick to only one page. If both pages are open simultaneously then referencing will be easy. The entire related link can be easily seen.

V. CONCLUSION

From the above proposed behavior an application framework can be developed showing two pages at a time. Here the human tendency of referring to multiple sources at a time is considered. The page is loaded at the user request and the follow on. The algorithm is applied in the framework for the evaluation of pages loading and displaying. So with respect to the content requested which is present in the server is downloaded to the client side through web browser. All these requires an active internet connection for the downloading the content, if the internet is not available the request page will not be downloaded or visible to the user. Again user has to reload the page for viewing. Current emerging technologies such as Web Sockets can be later used to have partial input available even when the complete vector (page) could not be received.). By spawning new tabs and keeping the existing ones open concurrently, users show relations between the pages.

REFERENCES

[1]   A. Aula, N. Jhaveri, and M. Käki, Information search and reaccess strategies of experienced web users. New York, New York, USA: ACM Press, 2005, pp. 583–592.
[2]   M. Viermetz, C. Stolz, V. Gedov, and M. Skubacz, "Relevance and Impact of Tabbed Browsing Behavior on Web Usage Mining," in 2006 IEEE/WIC/ACM Int. Conf. on Web Intelligence - WI'06, 2006, pp. 262–269.
[3]   F. Chierichetti, R. Kumar, and A. Tomkins, "Stochastic Models for Tabbed Browsing," in Proceedings of the 19th int. conf. on World wide web - WWW '10, 2010, pp. 241–250.
[4]   G. Singer, U. Norbisrath, E. Vainikko, H. Kikkas, and D. Lewandowski, "Search-logger analyzing exploratory search tasks," in Proc. of the 2011 ACM Symposium on Applied Computing - SAC'11, 2011, pp. 751–756.
[5]   J. Huang and R. W. White, "Parallel browsing behavior on the web," in Proc. of the 21st ACM conf. on Hypertext and hypermedia-HT'10, 2010, pp. 13–17.
[6]   J. Huang, T. Lin, and R. W. White, "No Search Result Left Behind: Branching Behavior with Browser Tabs," in Proc. of the 5th ACM int. conf. on Web search and data mining - WSDM'12, 2012, pp. 203–212.
[7]   P. Dubroy and R. Balakrishnan, "A Study of Tabbed Browsing Among Mozilla Firefox Users," in Proc. of the 28th int. conf. on Human factors in comp. systems - CHI '10, 2010, pp. 673–682.
[8]   H. Zhang and S. Zhao, "Measuring Web Page Revisitation in Tabbed Browsing," in Proc. of the 2011 annual conf. on Human factors in computing systems - CHI '11, 2011, pp. 1831–1834.