



Applying Web Usage Mining to a University Website Access Domain

Nirali Honest

Assistant Professor,
Smt. Chandaben Mohanbhai
Patel Institute of Computer
Applications,
CHARUSAT, Changa

Bankim Patel

Director,
Shrimad Rajchandra Institute of
Management and Computer
Application,
Uka Tarsadia University
Bardoli.

Atul Patel

Associate Professor,
Smt. Chandaben Mohanbhai
Patel Institute of Computer
Applications,
CHARUSAT, Changa

ABSTRACT

Web Usage Mining (WUM) is the process of taking out interesting behavior patterns that allow analyzing uses by the website administrator. In this paper we discuss the impact of developing the WUM process according to the requirements specific to the access of University website. WUM becomes a important aspect in today's era because the quantity of data is continuously increasing, the data resulted as part of web traffic comes from different sources and different format so issues related to integration and analyzing this heterogeneous and complex data are required before conducting any WUM analysis. Apart from these factors (*large size of data and Heterogeneous structure*) the three steps of WUM process are not coordinated to create a coherent and unique process. With these main concerns we decide to work for University website management domain and prepare a new reactive approach which uses the web usage data .site topology, academic calendar of university, in order to produce more specific process and results for University environment.

General Terms

Web Usage Mining, Web tools Analysis.

Keywords

Master Page concept, Data Pre-processing, Academic events.

1. INTRODUCTION

Since 90's, World Wide Web (WWW) has an immense speedy expansion. The WWW has become one of the most important medium to store and share disperses information. Apart from it the WWW is serving as a huge widely distributed global information service center for technical information, news, advertisement, e-commerce and other information service. But at the same time WWW is opening a dynamic and heterogeneous global distribution network. Due to huge information and lack of unified structure information retrieval is difficult. Most users may not have good knowledge of the structure of the information network, and may easily get fed up by taking many access hops and losing impatience when waiting for the information [1]. These challenges will have been solved efficiently by Web mining, which is the application of data mining technologies in Web. Web mining can extract useful and interesting pattern and potential knowledge from relevant record. Web mining consists of three active research areas: Web content mining, Web structure mining and Web usage mining. To meet the challenges of rapid expansion of the WWW one can study user and system behavior by exploring Web access logs.

Web is the single largest data source in the world, due to heterogeneity and lack of structure of web data, mining is a challenging task [2]. Web offers an unprecedented opportunity and challenge to data mining, because the amount of information on the Web is voluminous and diverse, Information/data of almost all types exist on the Web, e.g., texts, multimedia data, etc., Much of the Web information is semi-structured due to the nested structure of HTML code. This data must be accumulated into a reliable, integrated and completed order to be used for further phases of WUM.

In this paper, we are concerned about Web Usage Mining (WUM), also named as Web log mining. WUM focus on analyzing visiting information from logged data in order to extract usage pattern. These usage patterns can be used to improve Web server system performance and enhance the quality of service to the end users. In web mining, web log mining is high up due to effective use in numerous web related applications. These applications may include modification of web site design, schema modifications, web site and web server performance, fraud detection and future prediction, etc.

The main part of the paper is to focus the web page designed with the concept of Master page and the problem faced due to it and to focus on specifying a facility for adding an event for which the specific patterns can be generated. To consider these two points a University Website Access Domain (UWAD) is considered and explained in the later sections.

The remainder of this paper is organized as follows. In the section 2, Literature survey is discussed in brief. In the section 3, Motivation scenario is considered. Section 4 describes the concept of Master Page used in development of websites. Section 5 describes the Web Usage Mining and University website access domain (UWAD). Finally, the last section is conclusion.

2. LITERATURE SURVEY

The focus of literature review is to study the existing web usage mining areas and the preprocessing phase. Based on the purpose and application for the mining the entire process of web usage mining can be decided and the preprocessing phase can include or exclude the various steps. The study includes analysis, comparison and contrast of the available preprocessing techniques .How we can be more focused and guided at preprocessing level. Due to large amount of irrelevant entries in the web log file, the original log file cannot be directly used in WUM process. Therefore, the preprocessing of web log file becomes significant.



Jaideep Srivastava, et al. [3] provides a detailed taxonomy of the work in area of web mining, including research area as well as commercial offerings. With the growth of Web-based applications, specifically electronic commerce, there is significant interest in analyzing Web usage data to better understand Web usage, and apply the knowledge to better serve users. This has led to a number of commercial offerings for doing such analysis. An up-to-date survey of the existing work is also provided for Web Usage Mining Research Projects and Products. Major application areas of Web Usage mining have been identified in detail. Finally, a brief overview of the WebSIFT system as an example of a prototypical Web usage mining system is given. The WebSIFT system is designed to perform Web Usage Mining from server logs in the extended NSCA format (includes referrer and agent fields). The preprocessing algorithms include identifying users, server sessions, and inferring cached page references through the use of the referrer field.

Doru Tanasa, et al. [4] focus on web server logs from several web sites, generally belonging to the same organization. An important organization might have several web servers for its web sites. A user navigates through all these servers transparently because the pages from different web servers are strongly interlinked. Visitors may not even notice that the web server has changed. However for Web Usage Mining analyst who perform a user centric analysis, this is important because they want to have a complete overview of the users behavior on the entire web site, thus the analyst must reassemble the users path through all the different web servers that they visited. The authors provide a solution to join all the log files and reconstitute the visit. The classical data preprocessing involves three steps data fusion which may consists of joining and anonymizing log files, data cleaning which consists of removing unwanted things and data structuration may include user identification, session identification, visits, episodes, etc., the authors added a new step called data summarization which allows the analyst to select only the information of interest based on storing and data generalization.

Fang Yuan, et al. [5] mainly focus on analyzing visiting information from logged data in order to extract usage pattern, which can be classified on to three categories: similar user group, relevant page group and frequency accessing paths. These usage patterns can be used to improve Web server system performance and enhances the quality of service to the end users. In order to carry out this process authors give the algorithms for knowing the website's topological structure, data cleaning, user identification, session identification and path complement. The paper presents preprocessing of the web server logs in order to ensure its reliability to use in data mining algorithm.

Mohd Helmy Abd Wahab, et al. [6] discusses on the types of logs (Transfer log, Agent log, Error Log, Referer Log), types of formats for logs (W3C extended log file format, Microsoft IIS log file format, and NCSA Common log file format). The paper shows the algorithm for reading and cleaning server logs, data transferring from log files to database for storage. The author have created this algorithms using Active server pages (ASP).

K. R. Suneetha, et al. [7] focuses on the analysis of website top errors, potential visitors of the site, etc. which help system administrator and web designer to improve their system by determining occurred system errors, corrupted and broken links by using web usage mining. The author gives the details if types of files based on the location of the file, which include Server side logs, Proxy side logs and Client side logs. The other details include, web log file structure, status codes

of HTTP and overview of preprocessing process. The author have collected the NASA log file of Jul,95, and duration for experiment is week (7 days), based on this various results are generated like Total number of IP addresses/day, Number of Unique users/day, Number of Hits/day, Total failures/day.

Dipa Dixit [8] discusses the basic Web Log Mining process which includes Data cleaning, User and Session identification and Path Completion. They describe two approaches for data preprocessing 1. Preprocessing using XML and 2. Preprocessing using Text files. Then they provide comparison and application of these approaches. In the first approach i.e. preprocessing using XML the Log records in the web log file are converted into DOM tree structure using XML parsers, so the benefit in this approach is that preprocessing stages are analyzed very well, as organized in DOM tree structure, But time taken for conversion of records from XML to Knowledge base is more. The XML approach is applicable when log file consists of more number of attributes describing usage profile of users (IIS, Extended file format). In the second approach i.e. using text file, benefit is that it takes less time to transfer records from text file to Knowledge base. But Understanding of each steps of preprocessing would be difficult for user because this approach demands analysis an knowledge of how weblog looks. Text file approach is applicable when web log file consist of very few attributes describing usage profile of users (Common Log Format)

Carlos G. Marquardt, et al. [9] discusses the impact of developing the pre-processing phase according to concepts, problems and goals specific to web based learning environments. They focus on reduction of preprocessing efforts by automation of a number of tasks, the easier alignment of mining goals with the results that can be obtained and the possibility of involvement of domain related people with less technical skills w.r.t. mining. They encouraged that depending on the goal pursued, new approaches new approaches more related to specifics of learning environments can lead to more interesting results. In this concern they detail the traditional task involved in pre-processing, describing established techniques, as well as new approaches that take into account concepts and criteria related to web based learning environment.

3. A MOTIVATION SCENARIO

In Web usage mining the three steps of WUM process are not coordinated to create a coherent and unique process, so as per the literature review, there are three main points for generation of this concept,

1. Web usage mining process can be molded according to the specific goal w.r.t mining.
2. There is no support for generation of reports for particular events, you need to remember the interval of the event for generating the report of the event. (i.e. specify date intervals every time for generating the reports.)
3. The websites designed by Content Management System, Master Page concept and where web pages are given unique page numbers that are fetched from database, the report generation for per page frequency is not supported by certain tools, and if supported the page name cannot be known if it is generated by the ID number.

We try to incorporate these features in our architecture, and to study these we consider a website for University domain, which is designed with CMS, Master Page concept where



each content page is given unique ID and the page content is stored in database. So we try to mold the WUM process based on the University Website Access Domain (UWAD) and generate standard reports for the website in addition to specify events and then generate reports. Events in context of University are Admission process, Recruitment process, Special Technical and Non-Technical events and registration for them, Convocation, etc.

Preprocessing being preliminary and essential step but rather ignored due to variations and limitations of web log files. A web log file, as an input to the preprocessing phase of WUM, large in size, contains number of raw and irrelevant entries and is basically designed for debugging purpose [10]. Consequently, web log file cannot be directly used in WUM process.

Preprocessing of log file is complex and laborious job and it takes 80% of the total time of web usage mining process as whole [11]. Weighing the pros and cons, we come to the conclusion that, we can not negate importance of preprocessing step in web usage mining. Paying due attention to preprocessing step, improves the quality of data [12], furthermore, preprocessing improves the efficiency and effectiveness of other two steps of WUM such as pattern discovery and pattern analysis.

4. MASTER PAGE CONCEPT

4.1 Concept of Master Pages in Development of Websites

Recently many websites incorporate the concept of Master Pages. Master pages allow you to create a consistent look and behavior for all the pages (or group of pages) in your web application. A master page provides a template for other pages, with shared layout and functionality. The master page defines placeholders for the content, which can be overridden by content pages. The output result is a combination of the master page and the content page. The content page contains the content you want to display. Master pages actually consist of two pieces, the master page itself and one or more content pages. [13]. ASP.NET master pages allow you to create a consistent layout for the pages in your application.

Example of Master Page creation in ASP.NET

Example when users request the content page, ASP.NET merges the pages to produce output that combines the layout of the master page with the content of the content page.

Master Page Example

```
<% @ Master %>
<html>
<body>
  <h1>Standard Header For All Pages</h1>
  <asp:ContentPlaceHolder id="CPH1" runat="server">
    </asp:ContentPlaceHolder>
</body>
</html>
```

The master page above is a normal HTML page designed as a template for other pages. The @ Master directive defines it as a master page. The master page contains a placeholder tag <asp:ContentPlaceHolder> for individual content. The id="CPH1" attribute identifies the placeholder, allowing many placeholders in the same master page. This master page was saved with the name "master1.master". The master page can also contain code, allowing dynamic content.[21]

Content Page Example

```
<% @ Page MasterPageFile="master1.master" %>
<asp:Content ContentPlaceHolderId="CPH1"
runat="server">
  <h2>Individual Content</h2>
  <p>Paragraph 1</p>
  <p>Paragraph 2</p>
</asp:Content>
```

The content page above is one of the individual content pages of the web. The @ Page directive defines it as a standard content page. The content page contains a content tag <asp:Content> with a reference to the master page (ContentPlaceHolderId="CPH1"). This content page was saved with the name "mypage1.aspx". When the user requests this page, ASP.NET merges the content page with the master page. The content text must be inside the <asp:Content> tag. No content is allowed outside the tag.

Content Page With Controls

```
<% @ Page MasterPageFile="master1.master" %>
<asp:Content ContentPlaceHolderId="CPH1"
runat="server">
  <h2>W3Schools</h2>
  <form runat="server">
    <asp:TextBox id="textbox1" runat="server" />
    <asp:Button id="button1" runat="server"
text="Button" />
  </form>
</asp:Content>
```

The content page above demonstrates how .NET controls can be inserted into the content page just like an into an ordinary page.

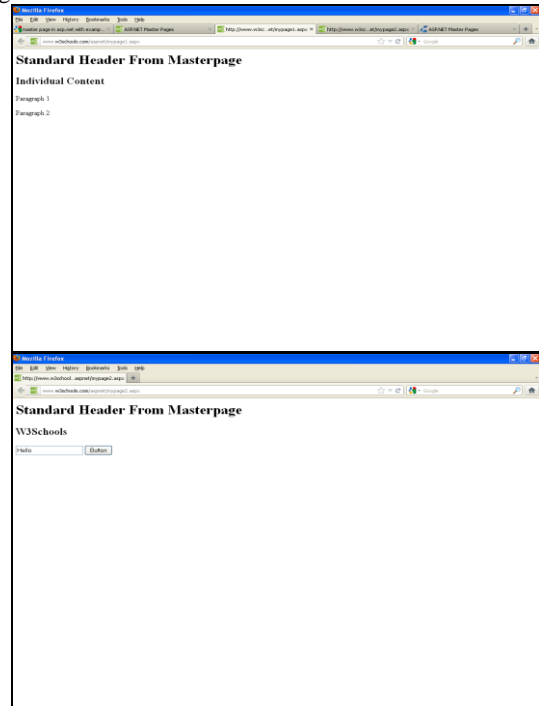


Fig 1 : Snapshot of Master and Content page

A master page is an ASP.NET file with the extension .master (for example, MySite.master) with a predefined layout that can include static text, HTML elements, and server controls. The master page is identified by a special @ Master directive that replaces the @ Page directive that is used for ordinary .aspx pages. The directive looks like the following.

```
<% @ Master Language="C#" %>
```

The @ Master directive can contain most of the same directives that a @ Control directive can contain. For example, the following master-page directive includes the name of a code-behind file, and assigns a class name to the master page.

```
<% @ Master Language="C#" CodeFile="MasterPage.master.cs" Inherits="MasterPage" %>
```

In addition to the @ Master directive, the master page also contains all of the top-level HTML elements for a page, such as html, head, and form. For example, on a master page you might use an HTML table for the layout, an img element for your company logo, static text for the copyright notice, and server controls to create standard navigation for your site. You can use any HTML and any ASP.NET elements as part of your master page.

Replaceable Content Placeholders

In addition to static text and controls that will appear on all pages, the master page also includes one or more ContentPlaceHolder controls. These placeholder controls define regions where replaceable content will appear. In turn, the replaceable content is defined in content pages. After we have defined the ContentPlaceHolder controls, a master page might look like the following.

```
<% @ Master Language="C#" %>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
```

```
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml" >
```

```
<head runat="server" >
```

```
<title>Master page title</title>
```

```
</head>
```

```
<body>
```

```
<form id="form1" runat="server">
```

```
<table>
```

```
<tr>
```

```
<td><asp:contentplaceholder id="Main" runat="server" /></td>
```

```
<td><asp:contentplaceholder id="Footer" runat="server" /></td>
```

```
</tr>
```

```
</table>
```

```
</form>
```

```
</body>
```

```
</html>
```

4.2 Content Pages

We can define the content for the master page's placeholder controls by creating individual content pages, which are ASP.NET pages (.aspx files and, optionally, code-behind files) that are bound to a specific master page. The binding is established in the content page's @ Page directive by

including a MasterPageFile attribute that points to the master page to be used. For example, a content page might have the following @ Page directive, which binds it to the Master1.master page.

```
<% @ Page Language="C#"
```

```
MasterPageFile="~/MasterPages/Master1.master"
```

```
Title="Content Page"%>
```

In the content page, you create the content by adding Content controls and mapping them to ContentPlaceHolder controls on the master page. For example, the master page might have content placeholders called Main and Footer. In the content page, you can create two Content controls, one that is mapped to the ContentPlaceHolder control Main and the other mapped to the ContentPlaceHolder control Footer, as shown in the following figure [22].

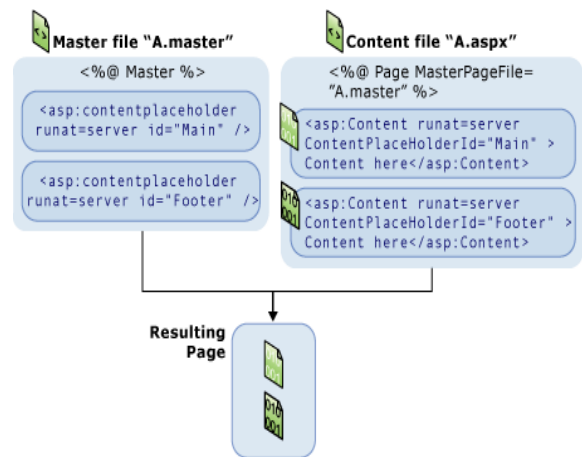


Fig 2 : Working of Master and Content Page

After creating Content controls, we add text and controls to them. In a content page, anything that is not inside the Content controls (except script blocks for server code) results in an error. We can perform any tasks in a content page that we do in an ASP.NET page. For example, we can generate content for a Content control using server controls and database queries or other dynamic mechanisms. A content page might look like the following. [code for C#]

```
<% @ Page Language="C#"
```

```
MasterPageFile="~/Master.master" Title="Content Page 1" %>
```

```
<asp:Content ID="Content1"
```

```
ContentPlaceHolderID="Main" Runat="Server">
```

```
Main content.
```

```
</asp:Content>
```

```
<asp:Content ID="Content2"
```

```
ContentPlaceHolderID="Footer" Runat="Server"
```

```
>
```

```
Footer content.
```

```
</asp:content>
```

The @ Page directive binds the content page to a specific master page, and it defines a title for the page that will be merged into the master page. Note that the content page



contains no other markup outside of the Content controls. (The master page must contain a head element with the attribute runat="server" so that the title setting can be merged at run time.)

We can create multiple master pages to define different layouts for different parts of your site, and a different set of content pages for each master page.

4.3 Advantages of Master Pages

Master pages provide functionality that developers have traditionally created by copying existing code, text, and control elements repeatedly; using framesets; using include files for common elements; using ASP.NET user controls; and so on. Advantages of master pages include the following:

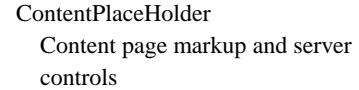
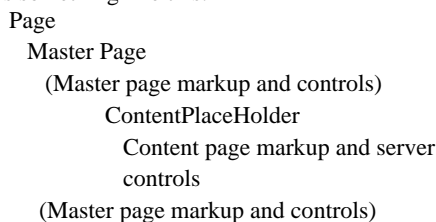
- They allow you to centralize the common functionality of your pages so that you can make updates in just one place.
- They make it easy to create one set of controls and code and apply the results to a set of pages. For example, you can use controls on the master page to create a menu that applies to all pages.
- They give you fine-grained control over the layout of the final page by allowing you to control how the placeholder controls are rendered.
- They provide an object model that allows you to customize the master page from individual content pages.

4.4 Run-time Behavior of Master Pages

At run time, master pages are handled in the following sequence:

1. Users request a page by typing the URL of the content page.
2. When the page is fetched, the @ Page directive is read. If the directive references a master page, the master page is read as well. If this is the first time the pages have been requested, both pages are compiled.
3. The master page with the updated content is merged into the control tree of the content page.
4. The content of individual Content controls is merged into the corresponding Content Placeholder control in the master page.
5. The resulting merged page is rendered to the browser.

From the user's perspective, the combined master and content pages are a single, discrete page. The URL of the page is that of the content page. From a programming perspective, the two pages act as separate containers for their respective controls. The content page acts as a container for the master page. The master page becomes a part of the content page. In effect, the master page acts in much the same way a user control acts — as a child of the content page and as a container within that page. In this case, however, the master page is the container for all of the server controls that are rendered to the browser. The control tree for a merged master and content page looks something like this:



(Master page markup and controls)

This diagram is simplified; if the content page does not have corresponding Content controls, the master page might also have markup and controls in the ContentPlaceholder controls. In general, this structure has no effect on how we construct our pages or program them. However, in some cases, if we set a page-wide property on the master page, it can affect the behavior of the content page, because the master page is the closest parent for the controls on the page.

4.5 Master Page and Content Page Paths

When a content page is requested, its content is merged with the master page, and the page runs in the context of the content page. For example, if you get the CurrentExecutionFilePath property of the Http Request object, whether in content page code or in master page code, the path represents the location of the content page. The master page and content page do not have to be in the same folder. As long as the MasterPageFile attribute in the content page's @ Page directive resolves to a .master page, ASP.NET can merge the content and master pages into a single rendered page.

4.6 Website Architecture

There are many websites which may use the concept of Master pages and develop the web pages, one of the architecture followed is that the content pages are given unique identifiers and they are fetched from the database, the pages don't have individual page names.

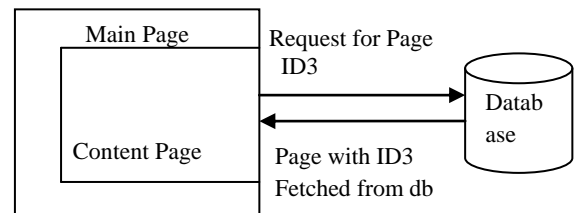


Fig 3: A CMS based web page content access pattern of Web pages having Unique ID

Due to this organization of architecture the URL generated while access of the pages is like below,

http://www.sitename.domainname/folder_UI/ContentPage.aspx?ID=1

The problem with current tools is as follows,

1. Certain tools don't generate per page hits of the pages organized by ID number, unless the user has directly accessed the website.
2. In the standard reports generated by various tools there is no mapping between ID number and name, so it is difficult for the website administrator to understand the report generated.

Example:

Page	Visits
/folder_UI/ContentPage.aspx?ID=10	250

Here the report addresses the Frequency of page visited, but it displays the ID number and not the Page name so it is not understood at the best.

3. There is no facility to generate events for particular actions. (i.e. Academic events like Admission Process, Recruitment Process, etc.)



5. WEB USAGE MINING AND UNIVERSITY WEBSITE ACCESS DOMAIN (UWAD)

WUM is the process that examines users' click stream [14][15][3]. The objective is to extract patterns that provide insights about users' navigation behavior over a Web environment. From a designer point of view, the structure of a Web site (contents and references between pages) reflects the expected site usage by its visitors [16].

The molding of WUM process for UWAD has major motives which include, Reduction of pre-processing efforts by the automation of a number of tasks, Easier association of mining goals with the results that can be obtained and Possibility of involvement of people having less technical skills w.r.t mining.

A university website is designed as a set of pages that constitute the Institute details, news and announcements, Listing of major actions like admission, recruitment, technical and nontechnical events, workshops and seminars, etc. By extracting access patterns of users behavior, various types of analysis can be developed over the given period which includes the listing of errors occurred while accessing the website, per page frequency, hit ratio for the given interval, maximum referred page, minimum referred page, first page accessed by the user, last page accessed by the user, navigation path of the user, hit ratio for special events like admission process, convocation, recruitment, etc.

The UWAD can certainly benefit from insights provided by WUM, but it is necessary to incorporate specific concepts and procedures from the context in study. According to Zaiane et al. [17][18], this transposition of characteristics to a new domain is not trivial, requiring that techniques and tools originally developed for the electronic commerce context (e.g. [19][20]) be customizable for adjusting to requirements of a new domain. For instance, they stress that the evaluation of Web-based learning environments using WUM techniques must be, at least partially, executable by domain-related people (e.g. instructors, site designers), who in general are not familiar with data mining techniques. There is no facility to generate events for particular action.

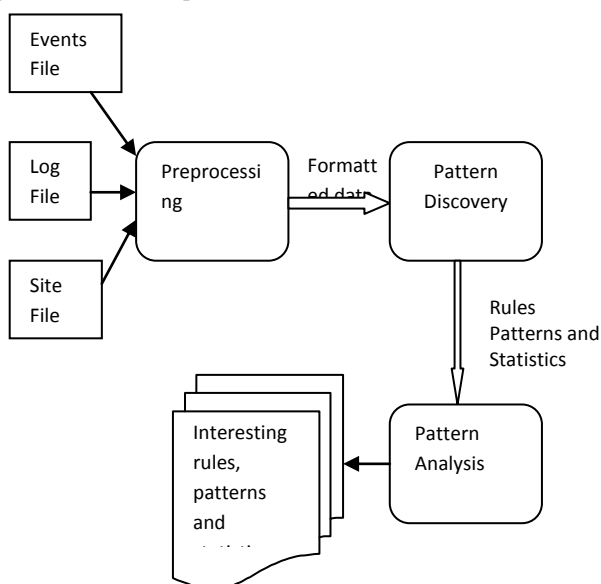


Fig 4: Architecture for UWAD

6. CONCLUSIONS

With these main concerns we decide to work for University website management domain and prepare a new reactive approach which uses the web usage data information, site topology, academic calendar of university, in order to produce more specific process and results for University environment. In general this concept may of use to any University website organization, which is follows a Master page layout concept in the website generation where pages are given unique IDs. Apart of this various things may be of help to the system administrator like, analysis of errors helps to know the problems while accessing the website, analysis of references to website during special event will help administrator to know and balance load, analysis of navigational patterns and duration will help the administrator with the knowledge about how to decrease the duration of the user by providing layout change, decrease in duration helps to usage of less bandwidth.

7. REFERENCES

- [1] HAN Jia-Wei, MENG Xiao-Feng, WANG Jing etc. Research on Web Mining. Journal of Computer ResearchLkDevelopment, 2001,38(4): 405-414.
- [2] Bing Liu, Web Content Mining, The 14th International World Wide Web Conference (WWW-2005), May 10-14, 2005, Chiba, Japan..
- [3] Jaideep Srivastava, Robert Cooley, Mukund Deshpande, Pang-Ning Tan, "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data", ACM SIGKDD Explorations, Volume 1, Issue 2, Jan 2000.
- [4] Doru Tanasa and Brigitte Trousse, "Advanced Data Preprocessing for Intersites Web Usage Mining", IEEE Computer Society, March/April, 2004.
- [5] Fang Yuan, Li-Juan Wang, Ge Yu, "Study on Data Preprocessing Algorithm in Web Log Mining", Proceedings of the Second International Conferences on Machine Learning and Cybernetics, Xi'an, 2-5 November 2003.
- [6] Mohd Helmy Abd Wahab, Mohd Norzali Haji Mohd, et. Al, "Data Pre-processing on Web Server Logs for Generalized Association Rules Mining Algorithm", World Academy of Science, Engineering and Technology, 2008.
- [7] K. R. Suneetha, Dr. R. Krishnamoorthi, "Identifying User Behavior by Analyzing Web Server Access Log File", International Journal of Computer Science and Network Security, Vol. 9 No. 4, April 2009.
- [8] Ms. Dipa Dixit and Ms M Kiruthika, "Preprocessing of web logs", International Journal on Computer Science and Engineering, Vol. 02, No. 07, 2010, 2447-2452.
- [9] Carlos G. Marquardt, Karin Becker and Duncan D. Ruiz, "A pre-processing tool for Web Usage Mining in the Distance Education Domain", Proceedings of the International Database Engineering and Applications Symposium (IDEAS'04) IEEE, 2004.
- [10] Khasawneh, N. and C.-C. Chan (2006). Active User-Based and Ontology-Based Web Log Data Preprocessing for Web Usage Mining. Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web



- Intelligence (WI 2006 Main Conference Proceedings) (WI'06) 0-7695-2747-7/06 © 2006.
- [11] Pabarskaite, Z. (2002). Implementing Advanced Cleaning and End-User Interpretability Technologies in Web Log Mining. 24th Int. Conf. information Technology Interfaces /TI 2002, June 24-27, 2002, Cavtat, Croatia.
- [12] Han, J. and M. Kamber (2006). Data Mining: Concepts and Techniques. A. Stephan. San Francisco., Morgan Kaufmann Publishers is an imprint of Elsevier.
- [13] Master Page Architecture and working, found at, <http://msdn.microsoft.com/en-us/library/wtxbf3hh.aspx>
- [14] B. Berendt and M. Spiliopoulou, Analysis of Navigation Behaviour in Web Sites Integrating Information Systems. VLDB Journal 9, 2000.
- [15] B. Mobasher, et. al, Automatic Personalization based on Web Usage Mining. In Communications of the ACM, 8, 2000.
- [16] M. Spiliopoulou, L. C. Faulstich, and K. Winkler, A Data Miner analysing the Navigational Behaviour of Web Users. In: Proc. of Workshop on Machine Learning in User Modelling , 1999.
- [17] O. R. Zaïane, Web Usage Mining for a Better Web-Based Learning Environment. Department of Computing Science University of Alberta Edmonton, Alberta, Canada, 2001.
- [18] O. R. Zaïane and J. Luo, Towards Evaluating Learners' Behaviour in a Web-Based Distance Learning Environment. In: Proceedings IEEE ICALT 2001, Madison, USA. 2001.
- [19] R. Cooley, et. al., Web Mining: Information and Pattern Discovery on the World Wide Web. 1997. Proc. IEEE Intl. Conf. Tools with AI, Newport Beach, CA, pp. 558-567, 1997.
- [20] R. Cooley, B. Mobasher, and J. Srivastava, Data Preparation for Mining World Wide Web Browsing Patterns. Journal of Knowledge and Information Systems, (1), 1999.
- [21] Master Page Information, found at , http://www.w3schools.com/aspnet/aspnet_masterpages.asp
- [22] Image Reference and path of the image <http://i.msdn.microsoft.com/dynimg/IC147301.gif>