

# A Visual System for Analyzing User Behaviour in Web Tasks

Gennaro Costagliola and Vittorio Fuccella  
Department of Mathematics and Informatics  
University of Salerno  
84084 Fisciano (SA) Italy  
{gencos, vfuccella}@unisa.it

## Abstract

*In this paper we present a visual system for the analysis of user behaviour during the execution of Web tasks based on visual data mining techniques. In particular, our system presents a suite of charts providing views of user behaviour starting from the well known clickstream visualization down to single user page interaction visualizations. The navigation through the views is enabled through ad-hoc defined zoom and filter operations. We give an overview of the architecture of the prototypical system and, in order to demonstrate the effectiveness of the approach and to give the reader an insight of the possible analysis types, we present a comparative analysis of an usability test performed on the two popular Web sites for booking flights.*

## 1 Introduction

The analysis of user behaviour during the accomplishment of Web navigation is a potentially fruitful source of important information. In particular, it has been extensively used for improving usability, but also to provide a better support to task users, i.e. by improving Web browsers and navigation tools, and to discover behavioural patterns in a given category of users.

Several approaches can be found in the literature that describe how to capture and visualize user behaviour in Web tasks. Most of them only rely on server log analysis. In recent years, there have been several attempts of integrating web server logs with client-side data, but only a few of them have resulted in an effective visual data mining method for user behaviour analysis. Among the earliest methods there are those exploiting *clickstream* visualization. As described in [6], *WebQuilt*, employs *clickstream* visualization for highlighting the most used paths by the users to accomplish a given web task. A variant of the above method, called *DiskTree*, transforms the *clickstream* graph in a 2D tree and visualizes it in a disk, such that the root

is located in the centre and the last level nodes on the circumference of the disk. In [3], Chi used this approach for discovering usability defects in Web sites. In [2], Chen et al. have improved this method by introducing operations and by describing a web graph algebra to manipulate and combine web graphs. Furthermore, specific approaches for the analysis of e-commerce systems capturing and visualizing user interface interactions are presented in [1, 4, 7].

In a previous work [5], we have described an approach for visually analyzing user behaviour in Web tasks. The approach visualizes a suite of charts providing views of user behaviour starting from the well known *clickstream* visualization down to single user page interaction visualizations. The navigation through the views is enabled through ad-hoc defined *zoom* and *filter* operations.

In this paper we present new issues related to the previous study: the primal approach has been enriched with a new chart type as a further level of analysis. Furthermore, we have integrated the suite of charts in a system, an *Integrated Analysis Environment*, which enables an easy browsing among analysis levels. Lastly, in order to demonstrate the effectiveness of the approach and to give the reader an insight of the possible analysis types, we have evaluated the usability of the Web-based interfaces for booking flights on two popular sites for vacation planning, *Expedia* and *Opodo*, making some comparisons among them.

The rest of the paper is organized as follows: The system is presented in Section 2. In Section 3, we discuss the case-study application. Several final remarks and a brief discussion on future work conclude the paper.

## 2 The system

In this section, we describe the system for the analysis of user behaviour during online tasks. The system is composed of two main components: a *Tracking Tool*, responsible for tracking user interactions and saving the recording in a log file, and a *Visualization Tool*, responsible for aiding the analyzer to graphically inspect the log.



Figure 1. Level 1 - Clickstream followed by task users for Opodo (a) and Expedia (b) Web sites.

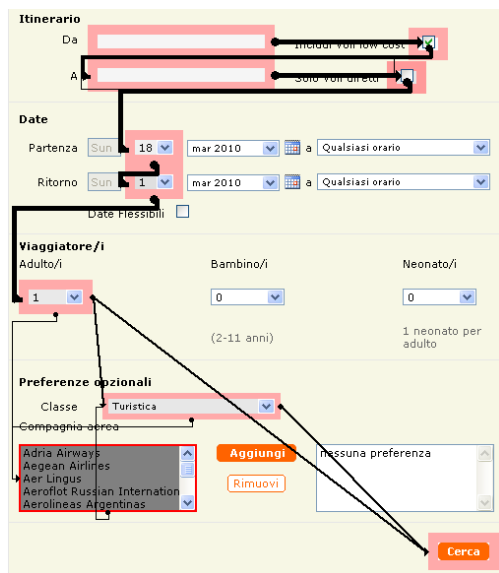


Figure 2. Second level chart for the first page of flight search procedure on Opodo Web site

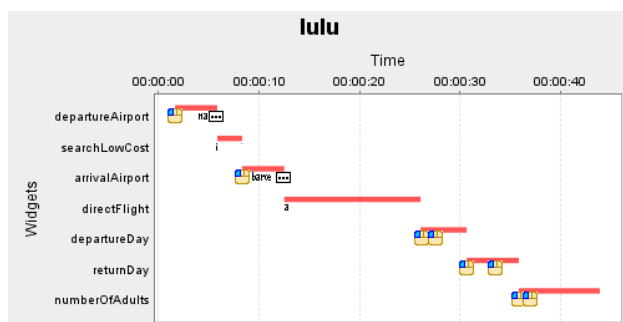


Figure 3. Third level chart of the fastest user to fill the form in the first page of flight search procedure on Opodo Web site

## 2.1 The Tracking Tool

The tracking tool relies on the AJAX technology in order to capture all of the users interactions with the user interface (running in the Web browser). It can be used as a framework to be instantiated in Web applications in order to track system users. Alternatively, it can be used as a stand-alone application (external to the tracked Web site) for tracking user behaviour on any Web site. The only requisite in the latter case, is to opportunely configure the Web browser to overcome security limitations, which do not allow the scripts to establish connections with external domains (cross-domain scripting security).

The tool is composed of a client-side and a server-side module. The former module is responsible for being aware of the behavior of the user while s/he is browsing the Web pages and for sending information related to the captured events to the server-side module. Gathering occurs by simply recording in an XML log file the interactions of the users with the Web site interface during navigation. The collected data carries information about the user session, the sequence of the pages loaded in the browser (*clickstream*) and the interactions of the users with the interactive elements contained in the pages, such as links and form fields.

## 2.2 The Visualization tool

The visualization tool is an *Integrated Analysis Environment* (IAE), which assists the user in the analysis of the Web tasks: it provides the user with the visualization of a suite of interactive charts, combined with textual information presented under the form of tables, to perform a cross-data analysis. The IAE has a Web-based interface, composed of three views:

1. In the centre of the screen there is the display, showing the current chart. The chart is interactive: by clicking with the mouse button on a sensitive area, it is possible to perform operations, zooming or filtering, or obtain detailed information in textual formats.

<b>Node border thickness</b>	number of sessions passing through the node
<b>Node border color</b> (in red scale)	avg num of passages through the node per session
<b>Node background color</b> (in gray scale)	avg time spent on the node
<b>Edge thickness</b>	num of sessions performing the transition
<b>Edge color</b> (in gray scale)	avg num of passages through the edge per session

**Table 1. Visual cues in first and second level charts**

2. On the left of the screen there is a toolbox view. A control panel allows to perform the same operations described above and more. In particular, it contains a list showing the sessions in the task. The currently selected ones are highlighted. A new selection can be performed by checking the sessions to show.
3. On the bottom of the screen there is a tabular view of the data. The data are selected and refreshed by interacting with the other views.

Charts enable analysis at three different levels. The **first level chart** represents *Clickstream* through a direct graph. The node set of the graph includes all the pages visited by at least one user while each edge is a transition between two pages. As shown in Figure 1(a), each node is represented through a thumbnail of the associated page.

The **second level chart** is a deepening of the previous. It shows information regarding a visit of a single page as shown in Figure 2. The used visual metaphore is a graph, as before. Nevertheless, this time the nodes represent the widgets (generally links and form fields) while the edges represent transitions of focus between them. In both levels, specific behavioural metrics are represented by the same visual cues, as summarized in table 1.

The **third level chart** is actually a set of user page charts, each showing the interactions of a single user with the selected page. This allows also for a comparative analysis of different behaviours. A user page chart, as shown in Figure 3, has an horizontal timeline in which bands of colors indicate different selections of events in the event stream. Each band is associated to a widget of the page. The widgets are placed on the categorical (Y) axis from top to bottom following the tab-order with which they are located in the page. Icons are placed inside the bands, representing user interactions with the widget at a given timestamp.

It is possible to go back and forth through the levels by applying *zoom (in and out)* operations. Furthermore, *filter* operations can be applied in order to construct the charts only by using a selection of the currently visualized user sessions involved in the task.

### 3 The case-study application

In order to demonstrate the effectiveness of the approach and to give the reader an insight of the possible analysis types, we have evaluated the usability of the Web-based interfaces for booking flights on two popular sites for vacation planning, Expedia and Opodo, making some comparisons among them. The following simple task has been administered to five students of the third year of the Laurea degree in Computer Science at the University of Salerno: “search on the Expedia (Opodo) Web site for the cheapest return flights for two adults from Napoli (Italy) to Barcelona (Spain), from november 11th to 18th, 2008”.

The test has been carried out in a laboratory equipped with PCs running a version of Firefox 3.x Web browser, opportunely configured to allow the interaction tracking. The task only lasted a few minutes (except the time necessary to configure the browser), and produced a log file sized less than 100Mb for recording the data for both the Web sites. The produced log has been analyzed with the *visualization tool*. The analysis has been performed independently for the two Web sites and the data have been compared later. For both of the interfaces, the analysis begins with the visualization of the first level chart (see Figure 1), which summarizes in a single view all of the sessions of the five users who performed the task.

It is worth noting that the two interfaces are similar. They are both composed of the following activities:

1. filling a form with travel information;
2. choosing the departure and return flights from a list of results;
3. checking the details of the choosen flight.

In the first level chart, graphs have the form of a chain, in which all the nodes have the same border color and thickness, and all the edges have the same color and thickness, meaning that each page was visited just once per session and all the users followed the same page sequences. Different node background colors stand for different average times spent on the pages.

The chain has three nodes for Opodo (Figure 1(a)) and five nodes for Expedia (Figure 1(b)), due to differences in the Web site interfaces: Opodo, by using AJAX, has a more compact structure. For Expedia, instead, the task requires more pages: two different pages to let user choose departure and return flights and a further page for disambiguating the names of departure and arrival airports. In our trials, by filling the departure field with the string “Napoli” Expedia detected an ambiguity between Napoli (southern Italian city) and Naples (a small town in Florida, USA).

By examining the times spent by the users on the pages (see tables 2 for Opodo and 3 for Expedia), we note that

User	Page1	Page2	Task
Alessia	59207	40064	99271
Gatto	92715	134388	227103
Lella	193387	119347	312734
Lulu	50026	47797	97823
Zanzi	119379	98629	218008
<b>Avg</b>	<b>102943</b>	<b>88045</b>	<b>190988</b>

**Table 2. Times (in milliseconds) spent by the task users on Opodo Web site**

User	Page1	Page2	Page3	Page4	Task
Alessia	61831	27729	15488	23850	128898
Gatto	82423	32853	11044	25734	152054
Lella	111734	28780	77957	92022	310493
Lulu	61580	32109	42962	12377	149028
Zanzi	109514	20198	31231	36274	197217
<b>Avg</b>	<b>85416</b>	<b>28334</b>	<b>35736</b>	<b>38051</b>	<b>187538</b>

**Table 3. Times (in milliseconds) spent by the task users on Expedia Web site**

Opodo required a longer time than Expedia, both for entering travel information (Page 1) and for choosing flights (Opodo, Page 2: about 88 seconds; Expedia, Page 3 and Page 4: about 74 seconds).

To have a more detailed insight into the situation we perform a *zoom* operation on the starting page of Opodo, obtaining the chart shown in Figure 2. The chart visualizes the behaviour of the users on the form. We note that most of the users interacted with the fields of the first two rows of the form following the tab-order. In particular, all of them interacted with the “include low cost flights” and “direct flights only” checkboxes on top-right of the image after filling the departure and arrival airport fields, respectively. This was due to the Opodo procedure for assisting the user in the input of the airport city names. The AJAX-based procedure does the following:

1. automatically completes the name while the user is typing;
2. moves the focus to the following form field.

We argue that the execution of the above procedure has disorientated the users when automatically moving the focus, by compelling them to interact with two unrequired fields (the checkboxes). This is much more evident by analyzing the third level charts showing the interactions of single users with the form. The chart of the fastest user to fill the form is reported in Fig. 3). The chart shows that the automatic completion (represented through an icon with three dots in a row) on the two text fields, *departureAirport* and *arrivalAirport*, has played an important role in speeding up the interaction of the user. Nevertheless, the moving of the focus to the following field, has confused the user.

The keypress events (represented through the pressed letters) highlighted on the bands representing the checkboxes, *searchLowCost* and *directFlight*, shows that the users were still typing the city name while the focus was already on the checkbox.

We deduce that many users looked at the keyboard instead of the screen when typing and did not realize that they were interacting with a new form field. Analyzing the chart, we note that the user wasted more than 15 seconds dealing with this tricky interface. The charts related to the other users show similar behaviours.

## 4 Conclusions

We have presented an approach for the analysis of Web tasks by means of Information Visualization. The approach significantly improves previous ones based on *clickstream* visualization. We described our prototypal system and presented a case-study to show how our approach aids the analyst in detecting usability problems in Web tasks.

We believe that many more patterns can be discovered than those highlighted in this paper and that the approach can be used with more general objectives other than evaluating Web usability. Thus, we are planning to perform further experiments, aimed at discovering and classifying new Web usage patterns and at testing our approach in different case-studies.

## References

- [1] J. Brainard and B. Becker. Case study: E-commerce clickstream visualization. In *IEEE Proceedings of Information Visualization 2001 (INFOVIS'01)*, pages 153–156, Washington, DC, USA, 2001. IEEE Computer Society.
- [2] J. Chen, L. Sun, O. R. Zaïane, and R. Goebel. Visualizing and discovering web navigational patterns. In *WebDB '04: Proceedings of the 7th International Workshop on the Web and Databases*, pages 13–18, New York, NY, USA, 2004. ACM.
- [3] E. H. Chi. Improving web usability through visualization. *IEEE Internet Computing*, 6(2):64–71, 2002.
- [4] T. Cofino, S. Gomory, J. Lee, and M. Podlaseck. Method for graphically representing clickstream data of a shopping session on a network with a parallel coordinate system. U. S. Patent Office, 2007. Patent number 7266510.
- [5] G. Costagliola and V. Fuccella. Fine-grained analysis of web tasks through data visualization. In *ICWE'09: Proceedings of the Ninth International Conference on Web Engineering*, June 2009 - to be printed.
- [6] J. I. Hong, J. Heer, S. Waterson, and J. A. Landay. Webquilt: A proxy-based approach to remote web usability testing. *ACM Transactions on Information Systems*, 19:263–285, 2001.
- [7] J. Lee and M. Podlaseck. Using a starfield visualization for analyzing product performance of online stores. In *EC '00: Proceedings of the 2nd ACM conference on Electronic commerce*, pages 168–175, New York, NY, USA, 2000. ACM.