

# Improving CADAL Portal Usability: Book Search, Reading, Reporting Services

Yin Zhang<sup>1</sup>, Jing Pan<sup>2</sup>, Jiangqin Wu<sup>1</sup> and Yueting Zhuang<sup>1</sup>

<sup>1</sup> College of Computer Science, Zhejiang University, 38 Zheda Road,  
310027 HangZhou, China

<sup>2</sup> Zhejiang University Libraries, Zhejiang University, 38 Zheda Road,  
310027 HangZhou, China  
{zhangyin98, panjing0525, wujq, yzhuang}@zju.edu.cn

**Abstract.** CADAL has been open to the public for more than two years. We have received a lot of positive feedbacks to improve the usability of CADAL portal. In this paper, we present our works in improving the usability of book search, reading and reporting services in CADAL portal. First, we present a quick book search application whose effective hybrid ranking mechanism combines content similarities with reading tendencies mined from book click-through logs. Second, we utilize Ajax and Flex technologies to improve user experiences of book search, reading and reporting services. Moreover, the use of Flex framework in the book reading service helps in preventing automatic batch downloading of books. Finally, Flex-based book reporting service makes administrators easily learn the hottest books, reading patterns of users and possible crawlers. The actual operation of those services shows that log mining and Flex technologies indeed improve the usability of CADAL portal.

**Keywords:** Portal, Flex, Search, Reading, Reporting Service

## 1 Introduction

China-America Digital Academic Library (CADAL) is one of mass-digitization projects in China. Since August 2006, one million digitized books have been available to the public through the CADAL portal <http://www.cadal.zju.edu.cn>. There are 410,498 visits coming from 103 countries/territories during past eleven months. The number of registration users in CADAL portal is 63,892. Many users have provided valuable suggestions to improve the usability of CADAL portal, in particular the book search and reading services. We therefore have utilized the book click-through logs and RIA technologies to improve the usability of book search and reading services. From a perspective of portal administration, the usability of book reporting service is important to the portal administrators to grasp the reading patterns and potential crawlers with ease.

Users can search and read books of their interest through the book search and reading services in the CADAL portal. In particular, the effectiveness of metadata-based book search service is improved to a great degree by our book click-through

log-based ranking mechanism. The full-text indexing technique is utilized for improving the efficiency of the book search service. The reading service consists of several modules: book page reading pane, book review and rating module, related books module, customized tagging module and bookmarking module. One of compulsory requirements of the reading service is the prevention of automatic batch downloading of books. The introduction of anti-downloading facilities (i.e., Access Frequency Detection and Captcha) results in many complaints of lots of visitors to browse books in random. In order to reduce the inconvenience induced by Captcha system, we are developing the new reading interface based on Flex and SSL technology. Finally, the reporting service based on click-through logs of books can help portal administrators easily grasp the number of books read by each user in a fixed period (e.g., day, week, and month), the countries/regions from which users came, reading patterns (e.g., the depth/length per reading session) and the possible automatic crawlers.

The rest of the paper is organized as follows: Section 2 introduces the related work, i.e. portal usability and RIA technologies. Section 3 presents the architecture of book search service and key algorithms that improve its effectiveness and efficiency. Section 4 shows the design of the book reading service and key modules that impact the user experience. Section 5 introduces the design of book reporting service and reports the usage statistics of book reading service during past 11 months. Section 6 draws conclusions and presents future work directions.

## **2 Related Work**

Portals are a special kind of websites offering a blend of information, application and services, which are typically based on more advanced web technologies that go beyond simple HTML pages. Hence, there are more nontechnical and technical things to consider for the users and designers to work with. In this section, we introduce the related work in portal usability and RIA technologies to build an expressive HCI interface for portal services.

### **2.1 Portal Usability**

ISO 9241-11 [1] defines usability as “extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use”. Nielsen et al. [2] introduce their findings and guidelines based on empirical evidence mainly from their testing of 716 web sites with 2,163 users around the world. In order to meet basic reasonable user interface requirements, a portal should: employ standards-based mark-up and be accessible through any browser, be functional across standard monitor sizes, use short, meaningful and user-friendly URLs, provide speedy performance, even under heavy usage, provide a robust and easy customizable interface.

The quality of access to the portal and its effectiveness as a tool for working with information can be characterized by the criteria for usability indicator, (e.g., accessibility, navigability and layout). Public value concept [3] has been used to

analyze and systemize the characteristics of government portals, which resulted in specifying five major indicators for assessing regional government portals. Portals are a mixture of information, applications and services. Thus, portal usability is more than the usability and design of its parts. It has to care for more general issues of blending information, applications and services, as well as tailoring a portal to a specific user group or role. Nowadays portals tend to be constructed by means of portlets, for which a usability model has been developed [4] from four dimensionalities: understandability, learnability, customizability, and compliance. Blandford et al. [5] have investigated the scope and limitations of four usability-oriented design and evaluation techniques as applied to digital libraries. The usability group at Helsinki University of technology designed an asynchronous cross-border usability testing [6] that was executed in five European countries, and distilled the cultural anomalies that would have been incorporated to the analysis. Quinn et al. [7] have created two user-friendly prototypes for displaying children's picture books with rich illustrations in the International Children's Digital Library by magnifying just the text, without magnifying the entire page.

## 2.1 RIA Technologies

Although HTML, CSS and Javascript can be used for many wonderful things, they lack the capability of developing modern sites and applications. The differences of browsers results in many techniques and Javascript libraries developed to accommodate their differences, but those techniques are complex and just reduce this frustration to some degree. Animation, video and a number of other things are extremely difficult or impossible with HTML alone, due to that HTML were designed for hyperlinked documents rather than the extremely rich presentations whose experience matters. Adobe Flex/Flash and Microsoft Silverlight have been the mainstream technologies building a complex interface that's anywhere near as responsive as a window in a rich client application.

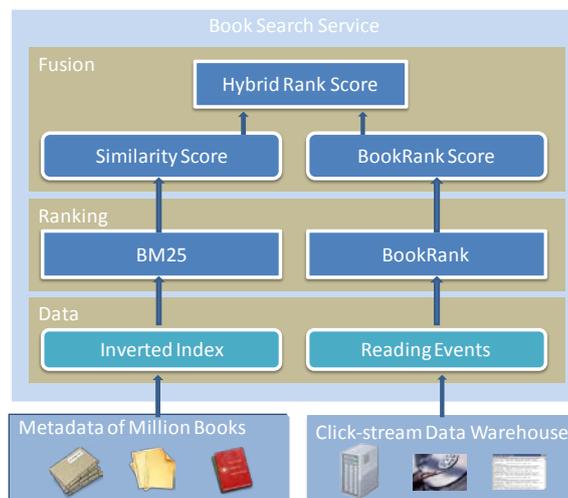
**Table 1.** The Comparisons between Adobe Flex/Flash and Microsoft Silverlight.

Capability	Adobe Flex/Flash	Microsoft Silverlight
Adoption	95%	25%
2D Drawing	Binary shape records	XAML
Media	H.263/264, ADPCM,MP3	VC-1,WMV,WMA
Animation	Transformation matrix (frame based)	WPF animation model (time based)
Programming Language	MXML, ActionScript	XAML, C#, VB etc.
Remote Messaging	AMF, Web Service	Web Service
Plugin Size	1.8MB	4MB
Tools	Flex Builder	VisualStudio,ExpressionBlend
UI	Dozens of Components, Layout Mechanisms	No built-in control

The Flex framework provides MXML, ActionScript, application services, components, and data connectivity to rapidly build rich internet applications [8]. MXML is the declarative language developers use to define the layout, appearance, and behaviors of a Flex application. ActionScript 3 is an object-oriented language that defines the client-side application logic. Application services include data binding, drag-and-drop management, display system, style system, and the effects and animation system. The component library provides all user interface controls, e.g., buttons, checkboxes, data grids, rich text editors, and containers developers use to design complex layouts with ease. For Flex remote messaging, developers can use open source BlazeDS project or Adobe LiveCycle Data Services.

Silverlight [9] is based on a scaled-down version of .NET's common language runtime (CLR) and thus allows developers to write client-side code using pure C#. The presentation system of Silverlight takes care of everything UI, including animation, text rendering, and audio/video playback, which can be accessed using Javascript DOM. Silverlight applications are created with a mixture of XAML, HTML, and javascript, so they are easy to integrate into existing web content. Table 1 shows the comparisons between Adobe Flex/Flash and Microsoft Silverlight described in [9][10].

### 3 Book Search Service

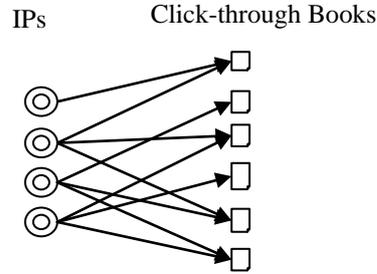


**Figure 1** System Architecture of Book Search Engine 2.0

In order to facilitate the search of books, we developed a book search engine for the CADAL portal in 2006, simply using the 'like' SQL statement in Microsoft SQLServer. Obviously, the old search engine is slow and ineffective due to the notoriously low efficiency of 'like' statement and the lack of effective ranking mechanisms. Hence, we developed and deployed the book search engine 2.0 by

utilizing the full-text indexing technique and log-based adaptive ranking mechanism for the CADAL digital library in 2008.

Figure 1 shows the overall architecture of the book search engine 2.0 for the CADAL portal. The architecture contains three layers: (1) Data layer (at the bottom): including the inverted index for the metadata of million books, and book reading events transformed from click-through logs; (2) Ranking layer: calculating similarity scores between query terms and metadata of books via a BM25 ranking function with field-specific BM25 parameters, and computing BookRank scores by applying a random walk-based algorithm [11] to the one-mode book correlation graph projected from the IP-Book bipartite graph built from reading events; (3) Fusion layer: combining similarity scores and BookRank scores to generate the final book scores, according to which the book ranking is generated. We implemented the index building and BM25 ranking modules based on Apache Lucene text search engine library. The book search engine 2.0 is deployed at the index page of CADAL portal.



**Figure 2** An IP-Book Bipartite Graph Transformed from a Sample Book Click-through Logs

The mechanism of book ranking is our major focus of the development of book search engine 2.0. We have implemented an adaptive book ranking approach via applying the random walks on the IP-Book bipartite graph (Fig. 2) transformed from the book Click-through Logs of the CADAL portal. Let  $U = \{u_i : 0 \leq i < m\}$  denote the set of unique IPs from which users visited CADAL portal for reading books. Let  $B = \{b_j : 0 \leq j < n\}$  denote the set of books read by all users from different IPs. Let  $T = \{t_{i,j} : u_i \in U \wedge b_j \in B, 0 \leq i < |U|, 0 \leq j < |B|\}$  denote the set of reading events in which book  $b_j$  was read by the user from unique IP  $u_i$ . Since users often browse several pages of books in random to look for ones of their interests, we identify a reading event by setting the threshold of the number of pages of the book read by the unique IP. The threshold of the number of book pages in a reading event is set to 20 in the current book search service, formally:

$$t_{i,j} = \begin{cases} True & \text{if } |\text{pages of book } b_j \text{ read by the unique ip } u_i| > 20 \\ False & \text{else} \end{cases}$$

We can construct an IP-Book bipartite graph (Fig.2) from those reading events, which were transformed from the book click-through logs. In order to discover the book ranking information on the IP-Book bipartite graph, we compress this bipartite graph into one-mode book graph whose nodes represent the corresponding books. The

weight of the edge between two nodes depends on the set of IPs co-visiting those two nodes. Formally

$$U_{i,j} = \begin{cases} \{u_k : (t_{k,i} \in T \wedge t_{k,i} = \text{True}) \wedge (t_{k,j} \in T \wedge t_{k,j} = \text{True})\} & \text{if } i \neq j \\ \emptyset & \text{if } i = j \end{cases}$$

where  $U_{i,j}$  denotes the set of IPs co-visiting two book  $b_i$  and  $b_j$ , it's obvious that  $U_{i,i}$  is the null set for the same book. In the following paragraphs, we use the respective adjacency matrix  $\tilde{C}$  to represent the one-mode book graph. The entry of the corresponding adjacency matrix is  $\tilde{C}_{i,j} = |U_{i,j}|$ , i.e., the number of IPs co-visiting the book  $i$  and  $j$ . To apply random walk with restart techniques on the IP-Book bipartite graph, we transformed the adjacency matrix into the column-stochastic matrix  $C$ , whose entry is  $C_{i,j} = \frac{\tilde{C}_{i,j}}{w_j}$ , where  $w_j = \sum_{0 \leq i < |B|} \tilde{C}_{i,j}$  is the sum of all entries of column  $j$ .

Next, we simulate random walks via the iterative multiplications of column-stochastic matrix with the new book ranking vector obtained in the previous iteration. To reflect the impact of historical book reading tendencies, we use the restart vector  $d$  to influence the final book ranking to some degree. The probability of transmitting into the adjacent books is  $\alpha$ . Thus, the probability of jumping to one of disconnected nodes is  $1-\alpha$ . Formally, computing the book ranking vector can be written

$$BR = \alpha C \cdot BR + (1-\alpha)d \quad (1)$$

where  $BR$  is the book ranking vector, its initial values  $BR(0)$  are set to uniform distribution. The values of  $BR(n)$  are updated after each iteration until convergence. Formally

$$\begin{cases} BR(0) = \frac{1}{|B|} \mathbf{1}_{|B|} \\ BR(n+1) = \alpha C \cdot BR(n) + (1-\alpha)d \end{cases} \quad (2)$$

In the end, we normalize the book ranking values as Eqn. (3), in order to enlarge the values of book ranking and then facilitate the linear combination of book ranking and content similarities.

$$br_i = \frac{br_i}{\max(BR)} \quad (3)$$

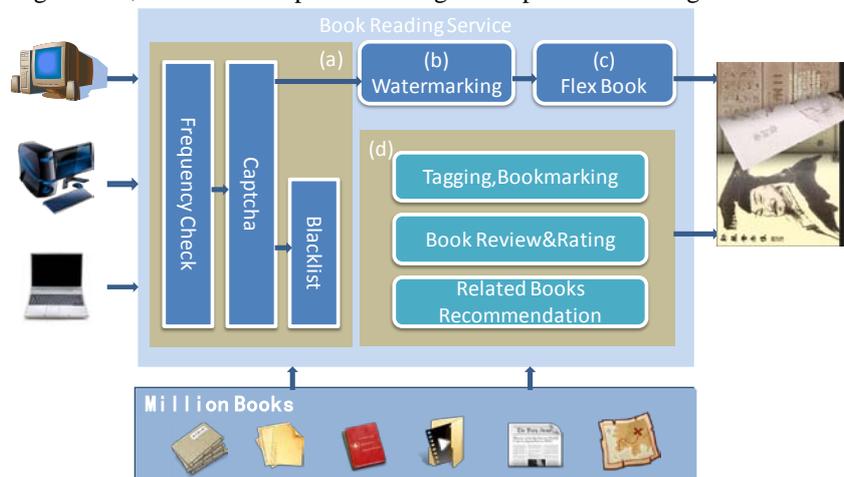
where  $\max(BR)$  is the largest ranking score in the ranking vector  $BR$ , obviously the normalized ranking score of the highest ranked book is 1.

We collected real book click-through logs from January to June in 2008, in which there are 71,076 books, 19,880 IPs and about 207,000 effective reading events. In the user study of just 5 persons, inspection of search results of two versions of book search engine demonstrates the utility of the book search engine 2.0. Experimental results show that our approach achieves more reasonable search results with higher probability of being hit, since our adaptive ranking mechanism reflects the reading trends of popularity on the whole. The book search engine 2.0 really benefits from

large volumes of usage data. Moreover, the use of full-text indexing technique greatly improves the efficiency of book search engine 2.0, compared to the ‘like’ statement used in book search engine 1.0. Now the response time per request is about 0.03 seconds. In the future, we plan to design a new kind of user interface for the next version of book search engine, in order to incorporate the multi-facet information of books. The users of the new multi-facet book search engine can search and browse the books of their interest simultaneously.

#### 4 Book Reading Service

One of main objectives of users visiting the CADAL portal is to read books of their interest. Hence, developing the user-friendly reading service is the key to improve the usability of the CADAL portal. Fig.3 shows the current internal modules in the book reading service of the CADAL portal. Fig. 3(a) illustrates the facilities of preventing automatic crawlers from downloading books in batch. Fig. 3(b) denotes the watermarking module. Fig. 3(c) is the flex book module with the support of book page flip effect, which removes the inconvenience of installing DjVu plugin by user themselves and provides user-friendly book reading experience. Fig. 3(d) shows the web-2.0 applications (i.e., tagging, bookmarking, reviewing and rating services) for the book reading service, in order to help users manage their personal reading.



**Figure 3** The Internal Modules of the Book Reading Service

The anti-crawler facilities (i.e., access frequency checking and Captcha Turing test) have severely hurt users’ book reading experience. Many users complain about the inconvenience of often being enforced to type in verification codes after quickly browsing dozens of pages of books in random. In order to lower the frequency of occurrence of Captcha Turing test, we are developing a new user interface of book reading service by utilizing the Flex/Flash technologies. Although the Flash/SWF format is an open format and ActionScript 3 codes in SWF can be extracted out, we

can encrypt and obfuscate the ActionScript 3 codes in the SWF files via encryption tools such as DoSWF, Amayeta SWF Encrypt. The encryption and obfuscation of ActionScript3 codes in SWF helps in hiding the URLs of book pages and program logics of how to compose the external URLs of book pages, compared to the HTML/Javascript codes the crawler can access at will. The Flex/Flash-based book reading service can help in strengthening the security of book data. Moreover, the Flex/Flash based user interface can support page flip effect via FlexBook and ImageZoom components, which improves a lot the reading experience (Fig. 4).

The current book reading service incorporates several personal service modules: tagging, bookmarking, book review&rating, related books recommendation. Web-2.0 applications (i.e. tagging, bookmarking and book review&rating modules) help users organize their favorite books and contents from multiple perspectives and express their opinions and multi-criteria ratings to books of their interests. We develop the new collaborative filtering techniques with respect to those multi-criteria ratings [12]. However, it's found that users were reluctant to provide the explicit multi-criteria ratings during the operation of CADAL portal, especially under the condition that users often anonymously access CADAL portal. We also integrate the related books recommendation module into the book reading service, which aims at helping users discover the related books of the book being read.

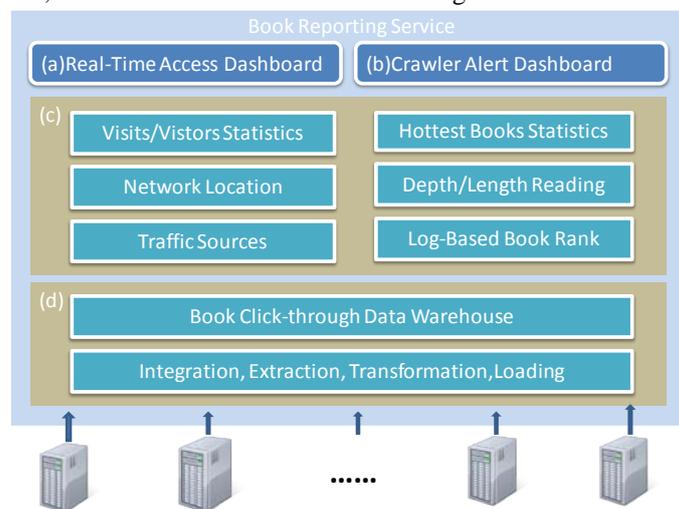


Figure 4 The Flip of Book Page in the Book Reading Service

## 5 Book Reporting Service

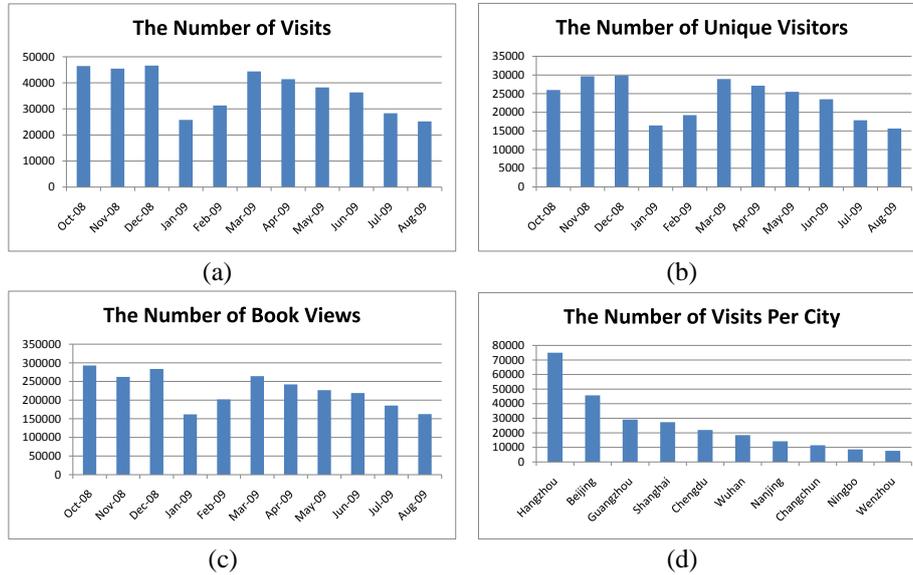
The book reporting service plays an important role in improving the usability of the CADAL portal. Fig. 5 shows the design of the book reporting service of the CADAL portal. Fig. 5(a)(b) represent the Flex-based user-friendly user interfaces to illustrate the real-time access information on the corresponding countries/regions of the global map, and possible automatic crawlers. The implementation of the real-time access dashboard is based on the IBM iLog Elixir visualization framework [13] and remote

messaging service capability of Flex framework [8]. Fig.5(c) shows the some statistics modules of the reporting service, e.g., statistics of visitors and visits per day, month and year, hottest books per repository, the distribution of traffic sources from which visitors find out CADAL portal, and network locations from which users came. Furthermore, we segment book click-through logs into many access sessions according to some criteria, or utilize book click-through logs to compute log-based book ranking in order to improve the usability of the book search service. In the future, we plan to implement the visualization and sequence analysis features proposed in [14][15]. Fig. 5(d) denotes the clickstream data warehouse [16], i.e., underpinning of the reporting service, which collects and integrates raw logs from distributed web servers, extracts out hit records of pages, transforms them into book reading events, and loads them into book click-through data warehouse.



**Figure 5** The Design Blueprint of the Book Reporting Service

Fig. 6 plots the results of the usage statistics of book reading events of the CADAL portal. Fig. 6(a) plots the bar chart of the number of visits per month from Oct. 2008 to Aug. 2009. Fig. 6(b) plots the bar chart of the number of unique visitors per month from Oct. 2008 to Aug. 2009. The number of visits and the unique visitors are both smallest in Jan. 2009, since the winter vacation of universities begins at the middle of Jan. 2009. Fig. 6(c) plots the bar chart of the number of book views per month from Oct. 2008 to Aug. 2009. The plot of book views is on the whole consistent with the plot of visits. Fig. 6(d) plots the bar chart of the number of visits during 11 months from ten cities in the descending order: Hangzhou, Beijing, Guangzhou, Shanghai, Chengdu, Wuhan, Nanjing, Changchun, Ningbo and Wenzhou. Table 2 summaries the countries/territories with the highest number of visits during the past 11 months. In table 2, the overwhelming majority of visits to the CADAL portal came from China. Moreover, there are nearly half new visits per month on average, no matter what country the visits of books originate from.



**Figure 6** Statistics of the Usage of Book Reading Service of the CADAL Portal

**Table 2.** Top 5 Network Locations with the Highest Number of Visits

Country/Territory	Visits	%New Visits
China	386,577	58.64%
Taiwan	5,623	36.40%
United States	4,646	42.40%
Hong Kong	4,203	51.01%
Japan	1,978	41.2%

## 6 Conclusions

CADAL portal has been a website with moderate traffic. Many advices for usability have been reported to portal administrators. We therefore utilize the Flex/Flash technology and book click-through logs to improve the usability of book search, reading and reporting services of the CADAL portal in terms of the effectiveness, efficiency and learnability of those services. The effectiveness of the book search service is improved by incorporating log-based book ranks into content similarities. After the use of inverted-index based full-text retrieval techniques, response time per query is about 0.03 seconds. Moreover, Flex/Flash technologies support the wonderful flip effect of book page in the book reading service, and strengthen the data safety measure of anti-downloading books in batch. In the book reporting service, Flex/Flash technologies help in plotting reading and crawling events real-timely on the dashboard, which alleviates the administration burden of portal administrators.

**Acknowledgments.** This work is supported by Development and Reform Committee under Grant No.1659[2004], and Program for Changjiang Scholars and Innovative Research Team in University (IRT0652).

## References

- [1] ISO9241-11, Ergonomic requirements for office work with visual display terminals (VDTs)-part 11: guidance on usability.
- [2] Neilsen, J., Loranger, H.(Eds.) *Prioritizing web usability*, New Riders Press, Berkeley CA, 2006.
- [3] Golubeva, A., Evaluation of Regional Government Portals on the Basis of Public Value Concept: Case Study from Russian Federation. In *Proceedings of the 1<sup>st</sup> International Conference on Theory and Practice of Electronic Governance*, Macao, China, 2007, 394-397.
- [4] Moraga, A., Calero, C., Piattini, M., Diaz, O., Improving a portlet usability model. *Software Quality Journal*, 2007, 15(2):155-177.
- [5] A. Blandford, S. Keith, I. Connell, H. Edwards, Analytical usability evaluation for digital libraries: a case study. In *Proceedings of the 4<sup>th</sup> ACM/IEEE-CS Joint Conference on Digital libraries*, Tuscon, AZ, USA, 2004, 27-36.
- [6] M. P. Nieminen, P. Mannonen, J. Viitanen, International remote usability evaluation: the bliss of not being there. *LNCS, Usability and Internationalization. HCI and Culture*, 2007, 388-397.
- [7] Quinn, A., Hu, C., Arisaka, T., Rose, A., Bederson, B., Readability of scanned books in digital libraries. In *Proceedings of the 26<sup>th</sup> annual SIGCHI conference on Human factors in Computing Systems*, Florence, Italy, 2008, 705-714.
- [8] <http://labs.adobe.com/technologies/flex/>
- [9] Introducing Microsoft Silverlight.  
<http://www.rau.ro/websites/e-society/lucrari/dragos%20pop%201.pdf>
- [10] Microsoft Silverlight Photography Framework, Comparing Component Based Designs in Adobe Flex and Microsoft Silverlight.  
<http://blog.davidroossien.com/softeng/papers/CS693/Silverlight%20Photography%20Framework.pdf>
- [11]Gori, M., Pucci, A., ItemRank: A Random-Walk Based Scoring Algorithm for Recommender Engines. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, Hyderabad, India, January 6-12, 2007, 2766-2771.
- [12]Yin Zhang, Yueting Zhuang, Jiangqin Wu, Liang Zhang. Applying probabilistic latent semantic analysis to multi-criteria recommender system. *AI Communications*, 2009, 22(2): 97-107.
- [13] <http://coenraets.org/blog/2009/05/tdfdashboard/>
- [14]Michael H., Linas Bukauskas, Arturas Mazeika, Peer Mylov. The 3DVDM Approach: A Case Study with Clickstream Data. In *Visual Data Mining-Theory, Techniques and Tools for Visual Analytics. LNCS 4404*, 2008.
- [15]Eric Lo, Ben Kao, Wai-Shing Ho, Sau Dan Lee. Chun Kit Chui, David W. Cheung. OLAP on sequences. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, Vancouver, Canada, 2008, 649-660.
- [16]Mark Sweiger, Mark Madsen, Jimmy Langston, Howard Lombard. *Clickstream Data Warehousing*. John Wiley&Sons, January, 2002.