



ELSEVIER

Available online at [www.sciencedirect.com](http://www.sciencedirect.com)



Library & Information Science Research 28 (2006) 407–432

**Library &  
Information  
Science  
Research**

# Search log analysis: What it is, what's been done, how to do it

Bernard J. Jansen

*College of Information Sciences and Technology, The Pennsylvania State University,  
329F IST Building, University Park, Pennsylvania 16802, USA*

---

## Abstract

The use of data stored in transaction logs of Web search engines, Intranets, and Web sites can provide valuable insight into understanding the information-searching process of online searchers. This understanding can enlighten information system design, interface development, and devising the information architecture for content collections. This article presents a review and foundation for conducting Web search transaction log analysis. A methodology is outlined consisting of three stages, which are *collection*, *preparation*, and *analysis*. The three stages of the methodology are presented in detail with discussions of goals, metrics, and processes at each stage. Critical terms in transaction log analysis for Web searching are defined. The strengths and limitations of transaction log analysis as a research method are presented. An application to log client-side interactions that supplements transaction logs is reported on, and the application is made available for use by the research community. Suggestions are provided on ways to leverage the strengths of, while addressing the limitations of, transaction log analysis for Web-searching research. Finally, a complete flat text transaction log from a commercial search engine is available as supplementary material with this manuscript.

© 2006 Elsevier Inc. All rights reserved.

---

## 1. Introduction

Researchers have used transaction logs for analyzing a variety of Web systems (Croft, Cook, & Wilder, 1995; Jansen, Spink, & Saracevic, 2000; Jones, Cunningham, & McNab,

---

*E-mail address:* [jjansen@acm.org](mailto:jjansen@acm.org).

0740-8188/\$ - see front matter © 2006 Elsevier Inc. All rights reserved.

doi:10.1016/j.lisr.2006.06.005

1998; Wang, Berry, & Yang, 2003). Web search engine companies use transaction logs (also referred to as search logs) to research-searching trends and effects of system improvements (cf. Google at <http://www.google.com/press/zeitgeist.html> or Yahoo! at [http://buzz.yahoo.com/buzz\\_log/?fr=fp-buzz-morebuzz](http://buzz.yahoo.com/buzz_log/?fr=fp-buzz-morebuzz)). Transaction logs are an unobtrusive method of collecting significant amounts of searching data on a sizable number of system users. However, there have been a limited number of researchers who explored transaction log methodology to study Web searching.

One possible reason is there are limited published works concerning how to conduct and employ transaction logs to support the study of Web searching, Web search engines, Intranet searching, or other Web-searching systems. This article addresses the use of transaction log analysis (also referred to as search log analysis) for the study of Web-searching and Web search engines in order to facilitate their use as a research methodology. A three-stage process composed of data *collection*, *preparation*, and *analysis* is presented for transaction log analysis. Each stage is addressed in detail and a stepwise methodology to conduct transaction log analysis for the study of Web searching is presented. A transaction log file is supplied as supplementary material to facilitate employment and experimentation with the analysis methodology. The strengths and shortcomings of transaction log analysis are presented. An application is offered that aids in supplementing transaction logs as a data collection method.

## 2. Review of the literature

### 2.1. What is a transaction log?

Not surprisingly, a transaction log is a file (i.e., log) of the communications (i.e., transactions) between a system and the users of that system. Rice and Borgman (1983) present transaction logs as a data collection method that automatically captures the type, content, or time of transactions made by a person from a terminal with that system. Peters (1993) views transaction logs as electronically recorded interactions between on-line information retrieval systems and the persons who search for the information found in those systems.

For Web searching, a transaction log is *an electronic record of interactions that have occurred during a searching episode between a Web search engine and users searching for information on that Web search engine*. A Web search engine may be a general-purpose search engine, a niche search engine, or a searching application on a single Web site. The users may be humans or computer programs acting on behalf of humans. Interactions are the communication exchanges that occur between users and the system. Either users or the system may initiate elements of these exchanges.

### 2.2. How are these interactions collected?

The process of recording the data in the transaction log is relatively straightforward. Web servers record and store the interactions between searchers (i.e., actually browsers on a

particular computer) and search engines in a log file (i.e., the transaction log) on the server using a software application. Thus, most transaction logs are server-side recordings of interactions. Major Web search engines execute millions of these interactions per day. The server software application can record various types of data and interactions depending on the file format that the server software supports.

Typical transaction log formats are access log, referrer log, or extended log. The W3C (<http://www.w3.org/TR/WD-logfile.html>) is one organizational body that defines transaction log formats. However, transaction logs for Web searching are a special type of transaction log file. This searching log format has most in common with the extended file format, which contains data such as the client computer's Internet Protocol (IP) address, user query, search engine access time, and referrer site, among other fields.

### 2.3. Why collect this data?

Once the server collects and records the data in a file, one must analyze this data in order to obtain beneficial information. The process of conducting this examination is referred to as *transaction log analysis* (TLA). TLA can focus on many interaction issues and research questions (Drott, 1998), but it typically addresses either issues of system performance, information structure, or measurements of user interactions.

In other views, Peters (1993) describes TLA as the study of electronically recorded interactions between on-line information retrieval systems and the persons who search for information found in those systems. Blečić et al. (1998) define TLA as the detailed and systematic examination of each search command or query by a user and the following database result or output. Phippen, Shepherd, and Furnell (2004) and Spink and Jansen (2004) also provide comparable definitions of TLA.

For Web-searching research, TLA is defined as *the use of data collected in a transaction log to investigate particular research questions concerning interactions among Web users, the Web search engine, or the Web content during searching episodes*. Within this interaction context, TLA could use the data in transaction logs to discern attributes of the search process, such as the searcher's actions on the system, the system responses, or the evaluation of results by the searcher.

The goal of TLA is to gain a clearer understanding of the interactions among searcher, content, and system or the interactions between two of these structural elements based on whatever research questions drive the study. From this understanding, one achieves some stated objective, such as improved system design, advanced searching assistance, or identified user information-searching behavior.

### 2.4. What is the theoretical basis of TLA?

TLA lends itself to a grounded theory approach (Glaser & Strauss, 1967). This approach emphasizes a systematic discovery of theory from data using methods of comparison and sampling. The resulting theories or models are grounded in observations of the "real world," rather than being abstractly generated. Therefore, grounded theory is an inductive approach to

theory or model development rather than the deductive alternative. For more on grounded theory see Chamberlain (1995).

Using TLA as a methodology, one examines the characteristics of searching episodes in order to isolate trends and identify typical interactions between searchers and the system. Interaction has several meanings in information searching, addressing a variety of transactions including query submission, query modification, results list viewing, and use of information objects (e.g., Web page, pdf file, video). Efthimiadis and Robertson (1989) categorize interaction at various stages in the information retrieval process by drawing from information-seeking research. TLA addresses levels one and two (*move* and *tactic*) of Bates' (1990) four levels of interaction, which are *move*, *tactic*, *stratagem*, and *strategy*. Belkin, Cool, Stein, and Theil (1995) have extensively explored user interaction based on user needs, from which they developed a multi-level view of searcher interactions. Saracevic (1997) views interaction as the exchange of information between users and system. Increases in interaction result from increases in communication content. Hancock-Beaulieu (2000) identifies three aspects of interaction, which are interaction within and across tasks, interaction as task sharing, and interaction as a discourse.

For TLA, interactions are *the physical expressions of communication exchanges between the searcher and the system*. For example, a searcher may submit a query (i.e., an interaction). The system may respond with a results page (i.e., an interaction). The searcher may click on a uniform resource locator (URL) in the results listing (i.e., an interaction). So, for TLA, interaction is a more mechanical expression of underlying information needs or motivations.

### 2.5. How extensively is TLA used?

Researchers and practitioners have used TLA to evaluate library systems, traditional information retrieval (IR) systems, and more recently Web systems. Peters (1993) provides a review of TLA in library and experimental IR systems. Some progress has been made in TLA methods since Peters' summary (1993) in terms of collection and ability to analyze data. Jansen and Pooch (2001) report on a variety of studies employing TLA for the study of Web search engines and searching on Web sites. Jansen and Spink (2005) provide a comprehensive review of Web-searching TLA studies.<sup>1</sup>

Employing TLA in research projects, Meister and Sullivan (1967) may be the first to have conducted and documented TLA results, and Penniman (1975) appears to have published one of the first research articles using TLA. There have been a variety of TLA studies since (cf. Baeza-Yates & Castillo, 2001; Chau, Fang, & Sheng, in press; Fourie & van den Berg, 2003; Millsap & Ferl, 1993; Moukdad & Large, 2001; Park, Bae, & Lee, 2005). Spink and Jansen (2004) provide an extensive bibliography of Web-searching TLA studies.

Discussing TLA as a methodological approach, Sandore, Flaherty and Kaske (1993) review methods of applying the results of TLA. Borgman, Hirsch, and Hiller (1996) comprehensively review past literature from different methodologies employed in these studies, including the

---

<sup>1</sup> Other review articles include Kinsella and Bryant (1987) and Fourie (2002).

goals of the studies. Several researchers have viewed TLA as a high-level designed process, including Copper (1998). Other researchers, such as Hancock-Beaulieu, Robertson, and Nielsen (1990), Griffiths, Hartley, and Willson (2002), Bains (1997), Hargittai (2002), and Yuan and Meadows (1999), have advocated using TLA in conjunction with other research methodologies or data collection. Alternatives for data collection include questionnaires, interviews, video analysis, and verbal protocol analysis.

Almost from its first use, researchers have critiqued TLA as a research methodology (Blecic et al., 1998; Hancock-Beaulieu et al., 1990; Phippen et al., 2004). These critiques report that transaction logs do not record the users' perceptions of the search, cannot measure the underlying the information need of the searchers, and cannot gauge the searchers' satisfaction with search results. Kurth (1993) comments that transaction logs can only deal with the actions that the user takes, not their perceptions, emotions, or background skills.

Kurth (1993) further identifies three methodological issues with TLA: *execution*, *conception*, and *communication*. Kurth (1993) states that TLA can be difficult to execute due to collection, storage, and analysis issues associated with the hefty volume and complexity of the data set (i.e., significant number of variables). With complex data sets, it is sometime difficult to develop a conceptual methodology for analyzing the dependent variables. Communication problems occur when researchers do not define terms and metrics in sufficient detail to allow other researchers to interpret and verify their results.

Certainly, any researcher who has utilized TLA would agree with these critiques. However, upon reflection, these are issues with many, if not all, empirical methodologies. Further, although Kurth's critique (1993) is still generally valid, advances in transaction logging software, standardize transaction log format, and improved data analysis software and methods have addressed many of these shortcomings.

As an additional limitation, transaction logs are primarily a server-side data collection method; therefore, some interactions' events are masked from these logging mechanisms, such as when the user clicks on the *back* or *print* button on the browser software, or *cuts* or *pastes* information from one window to another on a client computer. Transaction logs also, as stated previously, do not record the underlying situational, cognitive, or affective elements of the searching process.

In an effort to address these issues, Hancock-Beaulieu et al. (1990) developed a transaction logging software package that included online questionnaires to enhance TLA of browsing behaviors. This application was able to gather searcher responses via the questionnaires, but it also took away the unobtrusiveness (one of the strengths of the method) of the transaction log approach. Some software has been developed for unobtrusively logging client-side types of events, for example, the *Tracker* research package (Choo, Betlor, & Turnbull, 1998; Choo & Turnbull, 2000) and commercial spyware software systems.

In other tools for examining transaction log data, Wu, Yu, and Ballman (1998) present SpeedTracer, which is a tool for data mining Web server logs. However, given that transaction log data are usually stored in ASCII text files, relational databases or text-processing scripts work extremely well for TLA. Wang et al. (2003) used a relational database, as did Jansen et al. (2000) and Jansen, Spink, and Pederson (2005). Silverstein, Henzinger, Marais, and Moricz

(1999) apparently used text processing scripts. All approaches have advantages and disadvantages. With the text processing scripts, the analysis can be done in one pass. However, if additional analysis needs to be done, the whole data set must be re-analyzed. With the relational database approach, the analysis is done in incremental portions, but one can easily add additional analysis steps building off what has already been done.

### *2.6. How to conduct TLA for Web-searching research?*

Despite the abundant literature on TLA, there are little published manuscripts on how actually to conduct it. Some works do provide fairly comprehensive descriptions of the methods employed including Cooper (1998), Nicholas, Huntetymn, and Lievestey (1999), Wang et al. (2003), and Spink and Jansen (2004). However, none of these articles presents a process or procedure for actually conducting TLA in sufficient detail to replicate the method. This paper attempts to address this shortcoming.

## **3. TLA process**

TLA involves the following three major stages, which are as follows:

- collection: the process of collecting the interaction data for a given period in a transaction log;
- preparation: the process of cleaning and preparing the transaction log data for analysis; and
- analysis: the process of analyzing the prepared data.

Naturally, research questions need to be articulated, which determines what data need to be collected. However, transaction logs are typically of standard formats due to previously developed software applications. Given the interactions between users and Web browsers, which are the interfaces to Web search engines, the type of data that one can collect is standard. Therefore, the methodology provided with this manuscript is applicable to a wide range of studies.

### *3.1. Data collection*

The research questions define what information one must collect in a transaction log. Transaction logs provide a good balance between collecting a robust set of data and unobtrusively collecting that data. Collecting data from real users pursuing needed information while interacting with real systems on the Web affects the type of data that one can realistically assemble. If one is conducting a naturalistic study (i.e., outside of the laboratory) on a real system (i.e., a system used by actual searchers), the method of data monitoring and collecting cannot interfere with the information-seeking process. In addition to the loss of potential customers, a data collection method that interferes with the information-seeking process may unintentionally alter that process.

Table 1  
Snippet from a Web search engine transaction log

User identification	Date	Time	Search_url
ce00160c04c4158087704275d69fbedc	25/Apr/2004	04:08:50	Sphagnum Moss Harvesting+New Jersey+Raking
38f04d74e651137587e9ba3f4f1af315	25/Apr/2004	04:08:50	emailanywhere
fab953fe31996a0877732a1a970250a	25/Apr/2004	04:08:54	Tailpiece
5010dbbd750256bf4a2c3c77fb7f95c4	25/Apr/2004	04:08:54	l'personalities AND gender AND education'l
<b>25/Apr/2004</b>	<b>04:08:54</b>	<b>dmr panasonic</b>	
89bf2acc4b64e4570b89190f7694b301	25/Apr/2004	04:08:55	Bawdy poems
<b>397e056655f01380cf181835dfc39426</b>		<b>25/Apr/2004</b>	<b>gay porn</b>
a9560248d1d8d7975ffc455fc921cdf6a	25/Apr/2004	04:08:58	skin diagnostic
81347ea595323a15b18c08ba5167fbc3	25/Apr/2004	04:08:59	Pink Floyd CD label cover scans
3c5c399d3d7097d3d01aeea064305484	25/Apr/2004	04:09:00	freie stellen danggaard
9dafd20894b6d5f156846b56cd574f8d	25/Apr/2004	04:09:00	Moto.it
415154843dfe18f978ab6c63551f7c86	25/Apr/2004	04:09:00	Capability Maturity Model VS.
c03488704a64d981e263e3e8cf1211ef	25/Apr/2004	04:09:01	ana cleonides paulo fontoura

Note. Intentional errors are shown in boldface.

### 3.1.1. Fields in a standard transaction log

Table 1 provides a sample of a standard transaction log format collected by a Web search engine.

The fields are common in standard Web search engine transaction logs, although some systems may log additional fields. A common additional field is a cookie<sup>2</sup> identification code that facilitates identifying individual searchers using a common computer.

In order to facilitate valid comparisons and contrasts with other analysis, a standard terminology and set of metrics (Jansen & Pooch, 2001) is advocated, which will help address one of Kurth's critiques (1993) concerning the communication of TLA results across studies. Others have also noted terminology as issue in Web research (Pitkow, 1997). The standard field labels and descriptors are presented below.

A *searching episode* is a series of searching interactions within a given temporal span. Each record, shown as a row in Table 1, is a *searching interaction*. The format of each *searching interaction* is as follows:

- *User identification*: the IP address of the client's computer. This is sometimes also an anonymous user code address assigned by the search engine server, which is our example in Table 1.
- *Date*: the date of the interaction as recorded by the search engine server.
- *The time*: the time of the interaction as recorded by the search engine server.
- *Search URL*: the query terms as entered by the user.

<sup>2</sup> A cookie is a text message given by a Web server to a Web browser. The cookie is stored on the client machine.

Web search engine server software normally always records these fields. Other common fields include *Results Page* (a code representing a set of result abstracts and URLs returned by the search engine in response to a query), *Language* (the user preferred language of the retrieved Web pages), *Source* (the federated content collection searched), and *Page Viewed* (the URL that the searcher visited after entering the query and viewing the results page, which is also known as *click-thru* or *click-thorough*).

### 3.2. Data preparation

Once the data are collected, one moves to the data preparation stage of the TLA process. For data preparation, the focus is on importing the transaction log data into a relational database (or other analysis software), assigning each record a primary key, cleaning the data (i.e., checking each field for bad data), and calculating standard interaction metrics that will serve as the basis for further analysis.

Fig. 1 shows the Entity-Relation (ER) diagram for the relational database that will be used to store and analyze the data from our transaction log.

An ER diagram models the concepts and perceptions of the data and displays the conceptual schema for the database using standard ER notation. Table 2 presents the legend for the schema constructs names.

Since transaction logs are in ASCII format, one can easily import the data into most relational databases. A key thing here is to import the data in the same coding schema in which

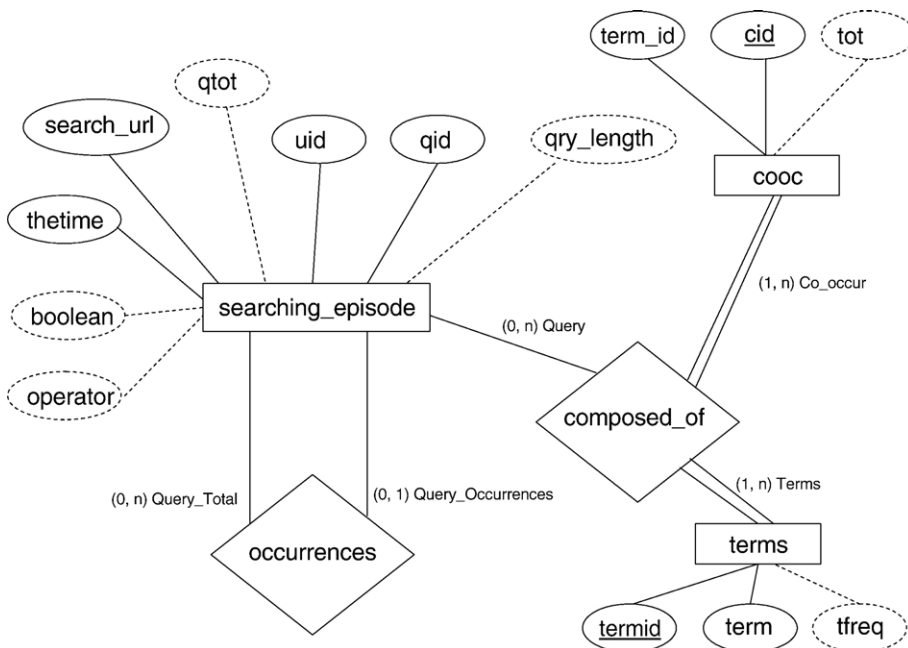


Fig. 1. ER scheme diagram Web searching transaction log.



Table 2  
Legend for ER schema constructs

Entity name	Construct
<i>Searching_Episodes</i>	A table containing the searching interactions
<i>boolean</i>	Denotes if the query contains Boolean operators
<i>operators</i>	Denotes if the query contains advanced query operators
<i>q_length</i>	Query length in terms
<i>qid</i>	Primary key for each record
<i>qtot</i>	Number of results pages viewed
<i>searcher_url</i>	Query terms as entered by the searcher
<i>thetime</i>	Time of day as measured by the server
<i>uid</i>	User identification based on IP
<i>Terms</i>	Table with terms and frequency
<i>term_ID</i>	Term identification
<i>term</i>	Term from the query set
<i>tfreq</i>	Number of occurrences of term in the query set
<i>Cooc</i>	Table term pairs and the number of occurrences of those pairs
<i>term_ID</i>	Term identification
<i>cid</i>	The combined term identification for a pair of terms
<i>tot</i>	Number of occurrences of pair in the query set

it was recorded (e.g., UTF-8, US-ASCII). Once imported, each record is assigned a unique identifier or primary key. Most modern databases can assign this automatically on importation, or one can assign it later using scripts.

### 3.2.1. Cleaning the data

Once the transaction log data are in a suitable analysis software package, the focus shifts to cleaning the data. Records in transaction logs can contain corrupted data. These records can result from multiple reasons, but they are mostly related to errors in logging the data. In the example shown in Table 1, one can easily spot these records (additionally these records are bolded), but many times a transaction log will number millions if not billions of records. So, a visual inspection is not practical for error identification. From experience, one method of rapidly identifying most errors is to sort each field in sequence. Since the erroneous data will not fit the pattern of the other data in the field, these errors will usually appear at the top of, bottom of, or grouped together in each sorted field. Standard data database functions to sum and group key field such as time and IP address will usually identify any further errors. One must delete all records with corrupted data from the transaction log database.

### 3.2.2. Parsing the data

Using the three fields of *The Time*, *User Identification*, and *Search URL*, common to all Web transaction logs, the chronological series of actions in a searching episode is recreated. The Web query transaction logs usually contain searches from both human users and agents. Depending on the research question, one may be interested in only human, common user terminals, or agent interactions. For the example in this manuscript, the interest is in only human-searching episodes. From the Web transaction log, a subset of interactions

must be culled that are deemed likely to have been submitted by humans. To do this, all sessions with less than 101 queries are separated into individual transaction logs for this research.

Given that there is no way to accurately identify human from non-human searchers (Silverstein et al., 1999; Sullivan, 2001), most researchers using a Web transaction log either ignore it (Cacheda & Viña, 2001) or assume some temporal or interaction cut-off (Montgomery & Faloutsos, 2001; Silverstein et al., 1999). Using a cut-off of 101 queries, the subset of the transaction log is weighted to queries submitted primarily by human searchers in a non-common user terminal, but 101 queries is also high enough not to introduce bias by too low of a cut-off threshold.

There are several methods to remove these large sessions. One can code a program to count the session lengths and then delete all sessions that have lengths over 100. For smaller log files (a few million or so records), it is just as easy to do with SQL queries. To do this, one must first remove records that do not contain queries. From experience, a transaction log may contain many such records as users go to Web sites for purposes other than searching.

### 3.2.3. Normalizing searching episodes

When a searcher submits a query, then views a document, and returns to the search engine, the Web server typically logs this second visit with the identical user identification and query, but with a new time (i.e., the time of the second visit). This is beneficial information in determining how many of the retrieved *results pages* the searcher visited from the search engine, but unfortunately, it also skews the results in analyzing how the user searched on the system.

So, one must separate these result page requests from query submissions for each searching episode. To do this the SQL query #00, Appendix A, can be used.

From a *tbl\_main*, this will create a new table *tbl\_searching\_episodes* with contains a count of multiple submissions (i.e., *qtot*) from each searcher within each record as shown in Fig. 2. This collapses the transaction log by combining all identical queries submitted by the same user to give the unique queries in order to analyze sessions, queries and terms, and pages of results (i.e., *tbl\_searching\_episodes*). Use the complete un-collapsed sessions (i.e., *tbl\_main*) in order to obtain an accurate measure of the temporal length of sessions. The *tbl\_searching\_episodes* will now be used for the remainder of our TLA. Use SQL query #01, Appendix A, to identify the sessions with more than 100 records. Then, one can delete these records from *tbl\_searching\_episodes* using the SQL delete query #02, Appendix A.

In TLA, there are many times one is interested in terms and term usage, which can be an entire study in itself. In these cases, it is many times cleaner to generate separate tables that contain each term and their frequency of occurrence. A term co-occurrence table that contains each term and its co-occurrence with other terms is also valuable for understanding the data. If using a relational database, one can generate these tables using scripts. If using text-parsing languages, one can parse these terms and associated data out during initial processing. We see these as *tbl\_terms* and *tbl\_cooc* in our database (see Fig. 1 and Table 2).

qid	uid	date	thetime	search_url	qtot	qry_length	boolean	operator
1	ce00160c04c415008704275d659becd	25/Apr/2004	04:08:50	Sphagnum Moss Harvesting + New Jersey + Raking	4	6	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	38f04d7e4e651137587e9ba34f1af515	25/Apr/2004	04:08:50	emailanywhere	2	1	<input type="checkbox"/>	<input type="checkbox"/>
3	fab953fe31996a0877732a1a970250a	25/Apr/2004	04:08:54	Tailpiece	1	1	<input type="checkbox"/>	<input type="checkbox"/>
4	5010dbbd750256b4a2c3c77b7e95c4	25/Apr/2004	04:08:54	1personalities AND gender AND education!1	1	5	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	daade90d883432d6c9b5609e3b41	25/Apr/2004	04:08:54	dmr panasonic	1	2	<input type="checkbox"/>	<input type="checkbox"/>
6	89b2acc4b64e4570b891907694b301	25/Apr/2004	04:08:55	bawdy poems"	1	2	<input type="checkbox"/>	<input type="checkbox"/>
7	96fa2a8d5a12a16380ed4ea1483b2b	25/Apr/2004	04:08:56	"Mark Twain"	1	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8	397e65665901380c1f181935d1c39426	25/Apr/2004	04:08:56	gay porn	1	2	<input type="checkbox"/>	<input type="checkbox"/>
9	a9590246d148479f59c455f921c0f6	25/Apr/2004	04:08:59	skin diagnostic	1	2	<input type="checkbox"/>	<input type="checkbox"/>
10	81347e695323a19518c038a451678a3	25/Apr/2004	04:08:59	Pink Floyd cd label cover scans	1	6	<input type="checkbox"/>	<input type="checkbox"/>
11	3c5c398d3d7097d3d1aee064305484	25/Apr/2004	04:09:00	feie stellen dangaard	1	3	<input type="checkbox"/>	<input type="checkbox"/>
12	9d9d20894b6d5f186846b56c45748d	25/Apr/2004	04:09:00	Moto.it	1	1	<input type="checkbox"/>	<input type="checkbox"/>
13	415154943d19679a6c635517c86	25/Apr/2004	04:09:00	Capability Maturity Model VS.	1	4	<input type="checkbox"/>	<input type="checkbox"/>
14	c03488704a64981e263e3e8cf1211ef	25/Apr/2004	04:09:01	ana cleonides paulo fontoura	1	4	<input type="checkbox"/>	<input type="checkbox"/>
15	7ab6e96ee504b65e47738c1f5999354	25/Apr/2004	04:09:01	quetschripen konstruktion kunststoff	2	3	<input type="checkbox"/>	<input type="checkbox"/>
16	013056951662fc13580e6e52d2a3b	25/Apr/2004	04:09:02	lovette password	3	2	<input type="checkbox"/>	<input type="checkbox"/>
17	eedec6d2ecc3519ea7747db982abf57	25/Apr/2004	04:09:04	free porn	1	2	<input type="checkbox"/>	<input type="checkbox"/>
18	3438445a250f377bf7bed41188a09b	25/Apr/2004	04:09:04	centro	2	1	<input type="checkbox"/>	<input type="checkbox"/>
19	e6871c24d889e9e8813cb361445040	25/Apr/2004	04:09:07	sex toys	1	2	<input type="checkbox"/>	<input type="checkbox"/>
20	cb06862ba4977b13161f1f0ea7e4ea4	25/Apr/2004	04:03:18	news "blue oak" "Guercus douglas"	1	5	<input type="checkbox"/>	<input checked="" type="checkbox"/>
21	9a44a9c5502a6b9ba9aff0e443245f9	25/Apr/2004	04:09:09	international investment on nanotechnology	4	4	<input type="checkbox"/>	<input type="checkbox"/>
22	e40da084ac6b776a3a57e8ccea5b62f	25/Apr/2004	04:09:10	mosquito keychain	2	2	<input type="checkbox"/>	<input type="checkbox"/>
23	fab953fe31996a0877732a1a970250a	25/Apr/2004	04:09:12	Valve Tailpiece	1	2	<input type="checkbox"/>	<input type="checkbox"/>
24	74574c974f301ea4b0561a00fab71b	25/Apr/2004	04:09:13	"wayne frocklage" Canadian pacific railway	1	5	<input checked="" type="checkbox"/>	<input type="checkbox"/>
25	143e599163fb75e16a52a6e050b01c	25/Apr/2004	04:02:46	matcho central	1	2	<input type="checkbox"/>	<input type="checkbox"/>
26	f69620a685279830416a6dd1afa0d73	25/Apr/2004	04:02:46	Pascal convert decimal to hex	2	5	<input type="checkbox"/>	<input type="checkbox"/>
27	7b080191cda654507f93842bd4b45	25/Apr/2004	04:02:46	bang bros "king chile"	1	4	<input type="checkbox"/>	<input checked="" type="checkbox"/>
28	89516e43dfwa2d4b7869ba8045514	25/Apr/2004	04:02:46	CFISCO	1	1	<input type="checkbox"/>	<input type="checkbox"/>
29	38e5d952e76d1c344977c27a914ea8	25/Apr/2004	04:02:47	"Stargate" desktop themes	1	3	<input type="checkbox"/>	<input type="checkbox"/>
30	dacc65152e9e0975623cb994bc0b447	25/Apr/2004	04:02:51	carmeland gallery	1	2	<input type="checkbox"/>	<input type="checkbox"/>
31	4167eb02168d2bc1698a196d7b10f65	25/Apr/2004	04:02:53	"femdom picture"	1	2	<input type="checkbox"/>	<input type="checkbox"/>
32	67b94c409c6046204894105830176ba	25/Apr/2004	04:02:55	"Carla Tenetti"	1	2	<input type="checkbox"/>	<input type="checkbox"/>
33	6b19010e49a39d7103eaab44530fb74	25/Apr/2004	04:02:56	efficacy	1	1	<input type="checkbox"/>	<input type="checkbox"/>
34	5a20077ed4f1d8a7b41de2acae6ad756	25/Apr/2004	04:02:58	http://kickme.to/aliasx	1	1	<input type="checkbox"/>	<input type="checkbox"/>
35	93cde4d894cc12cc85e27c4ad3626204	25/Apr/2004	04:03:14	P.M."	1	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>
36	1c1c9d058dbd43fadb3ee54851f9624	25/Apr/2004	04:03:01	quattro elementi	1	2	<input type="checkbox"/>	<input type="checkbox"/>
37	bae5967110a17a941254661k1a32050	25/Apr/2004	04:03:02	measurement high temp	1	3	<input type="checkbox"/>	<input type="checkbox"/>
38	606eb104503d0788e6f00385e58544a	25/Apr/2004	04:03:02	levicm watercube	2	2	<input type="checkbox"/>	<input type="checkbox"/>
39	0fcec5c427a7a1b93930538f5373904	25/Apr/2004	04:03:03	notebook mediamark	1	2	<input type="checkbox"/>	<input type="checkbox"/>
40	082206cc056dd658994304522400d4d	25/Apr/2004	04:03:03	www.dancentry.com/cicada/wedding.html	1	1	<input type="checkbox"/>	<input type="checkbox"/>
41	5b43061638aef9d59d351542c981fa	25/Apr/2004	04:03:04	mensajes subliminales un la publicidad"	1	5	<input checked="" type="checkbox"/>	<input type="checkbox"/>
42	07f54b225084bfe3c746c320222b5b	25/Apr/2004	04:03:13	online texting	1	2	<input type="checkbox"/>	<input type="checkbox"/>
43	07f54b225084bfe3c746c320222b5b	25/Apr/2004	04:03:04	onlinetexting	1	1	<input type="checkbox"/>	<input type="checkbox"/>
44	177219618313a1363469ba9d91d5a7c	25/Apr/2004	04:03:12	Art	1	1	<input type="checkbox"/>	<input type="checkbox"/>
45	177219618313a1363469ba9d91d5a7c	25/Apr/2004	04:03:14	Art history	1	1	<input type="checkbox"/>	<input type="checkbox"/>
46	cd7b9dc409c60de204894105830176ba	25/Apr/2004	04:02:57	"Jim Jansen"	1	2	<input type="checkbox"/>	<input type="checkbox"/>
						0	<input type="checkbox"/>	<input type="checkbox"/>

Fig. 2. Records of searching episodes with number of duplicate queries (qtot) recorded.

There are already several fields in our database, many of which can provide valuable information (see Fig. 1 and Table 2). From these items, one can calculate several metrics, some of which take a long time to compute for large data sets. Fig. 3 shows the cleaned and prepared database tables and relationships containing our transaction log ready for data analysis.

### 3.3. Data analysis

This stage focuses on three levels of analysis. These levels are discussed and the data analysis stage is stepped through.

### 3.4. Analysis levels

The three common levels of analysis for examining transaction logs are *term*, *query*, and *session*.

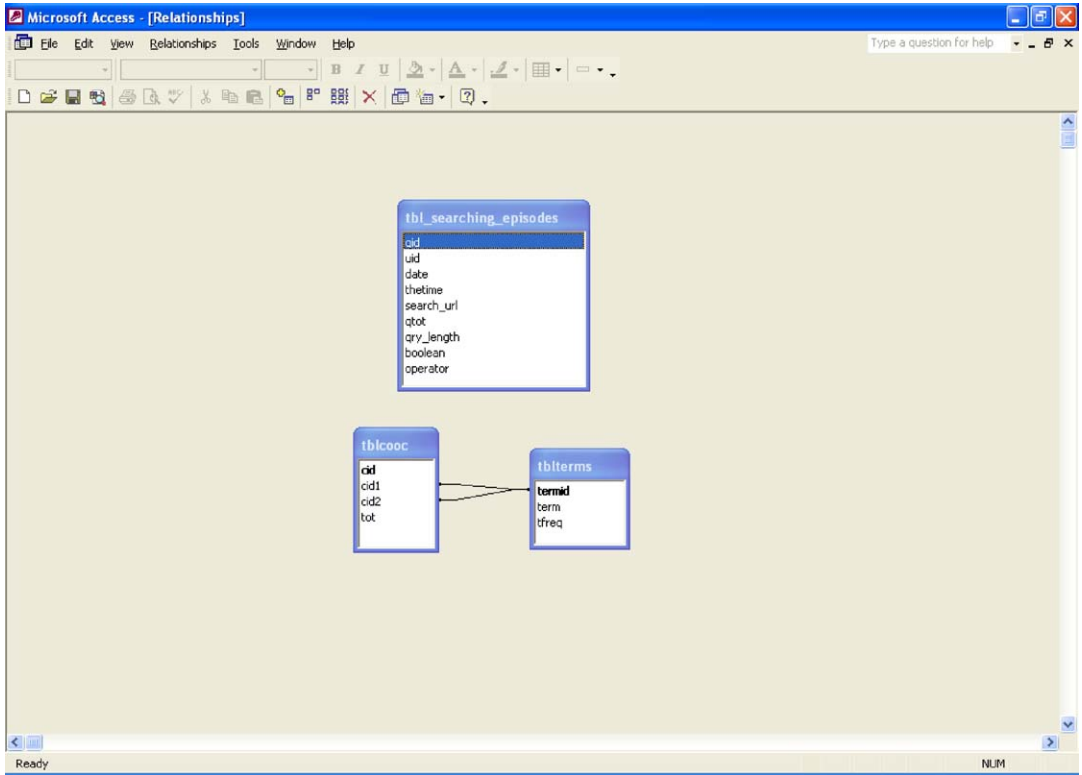


Fig. 3. Cleaned and prepared database of transaction log.

### 3.4.1. Term level analysis

The term level of analysis naturally uses the *term* as the basis for analysis. A *term* is a string of characters separated by some delimiter such as a space or some other separator. At this level of analysis, one focuses on measures such as *term occurrence*, which is the frequency that a particular term occurs in the transaction log. *Total terms* is the number of terms in the data set. *Unique terms* are the terms that occur in the data regardless of the number of times they occur. *High usage terms* are those terms that occur most frequently in the data set. *Term co-occurrence* measures the occurrence of term pairs within queries in the entire transaction log. One can also calculate degrees of association of term pairs using various statistical measures (cf. Ross & Wolfram, 2000; Silverstein et al., 1999; Wang et al., 2003).

### 3.4.2. Query level analysis

The query level of analysis uses the query as the base metric. A *query* is defined as a string list of zero or more terms submitted to a search engine. This is a mechanical definition as opposed to an information-seeking definition (Korfhage, 1997). The first query by a particular searcher is as an *initial query*. A subsequent query by the same searcher that

is different than any of the searcher's other queries is a *modified query*. There can be several occurrences of different modified queries by a particular searcher. A subsequent query by the same searcher that is identical to one or more of the searcher's previous queries is an *identical query*.

In many Web search engine transaction logs, when the searcher traverses to a new results page, this interaction is also logged as an *identical query*. In other logging systems, the application records the page rank. A results page is the list of results, either sponsored or organic (i.e., non-sponsored), returned by a Web search engine in response to a query. Using either *identical queries* or some results page field, one can analyze the result page viewing patterns of Web searchers.

One can examine other measures at the query level of analysis. A *unique query* refers to a query that is different from all other queries in the transaction log, regardless of the searcher. A *repeat query* is a query that appears more than once within the data set by two or more searchers.

*Query complexity* examines the query syntax, including the use of advanced searching techniques such as Boolean and other query operators. *Failure rate* is a measure of the deviation from the published rules of the search engine. The use of query syntax that the particular IR system does not support, but may be common on other IR systems, is *carry over*.

### 3.4.3. Session level analysis

At the session level of analysis, one primarily examines the within-session interactions (Hancock-Beaulieu, 2000). However, if the transaction log spanned more than one day or assigns some temporal limit to interactions from a particular user, one could examine between-sessions interactions. A *session interaction* is any specific exchange between the searcher and the system (i.e., submitting a query, clicking a hyperlink, etc.). A *searching episode* is defined as a series of interactions within a limited duration to address one or more information needs. This session duration is typically short, with Web researchers using between five and 120 minutes (cf. He, Göker, & Harper, 2002; Jansen & Spink, 2003; Montgomery & Faloutsos, 2001; Silverstein et al., 1999). The searcher may be multitasking (Miwa, 2001; Spink, 2004) within a searching episode, or the episode may be an instance of the searcher engaged in successive searching (Lin, 2002; Ozmutlu, Ozmutlu, & Spink, 2003; Spink, Wilson, Ellis, & Ford, 1998). This *session* definition is similar to the definition of a *unique visitor* that is used by commercial search engines and organizations to measure Web site traffic. The number of queries per searcher is the *session length*.

*Session duration* is the total time the user spent interacting with the search engine, including the time spent viewing the first and subsequent Web documents, except the final document. Session duration can therefore be measured from the time the user submits the first query until the user departs the search engine for the last time (i.e., does not return). This viewing time of the final Web document is not available since the Web search engine server does not record the time stamp. Naturally, the time between visits from the Web document to the search engine may not have been entirely spent viewing the Web document, which is a limitation of the measure.

A *Web document* is the Web page referenced by the URL on the search engine's results page. A Web document may be text or multimedia and, if viewed hierarchically, may contain a nearly unlimited number of sub-Web documents. A Web document may also contain URLs linking to other Web documents. From the results page, a searcher may click on a URL (i.e., visit) one or more results from the listings on the result page. This is *click through analysis* and measures the page viewing behavior of Web searchers. One measures *document viewing duration* as the time from when a searcher clicks on a URL on a results page to the time that a searcher returns to the search engine. Some researchers and practitioners refer to this type of analysis as *page view analysis*. *Click through analysis* is possible if the transaction log contains the appropriate data.

### 3.5. Conducting the data analysis

The key to successful TLA is conducting the analysis with an organized approach. One method is to sequentially number and label the queries (or coded modules) to correspond to the order of execution and to their function, since many of these queries must be executed in a certain order to obtain valid results. Many relational database management systems provide mechanisms to add descriptive properties to the queries. These can provide further explanations of the query function or relate these queries directly to research questions. Fig. 4 illustrates the application of such an approach.

Fig. 4 shows each query in sequence and provides a descriptive tag describing that query's function. To aid in reading, a list of queries is also provided in Appendix B.

One approaches TLA by conducting a series of standard analyses that are common to a wide variety of Web-searching studies. Some of these analyses may directly address certain research questions. Others may be the basis for more in-depth and further research analysis.

One typical question is "How many searchers have visited the search engine during this period?" One can determine this by using the SQL query #03, Appendix A. This query will provide a list of unique searchers and the number of queries they have submitted during the period. Naturally, a variety of statistical results can be determined using the previous queries. For example, one can determine the average number of queries per day using the SQL query #04, Appendix A.

One may want to know the session lengths for each searcher, which SQL query #05, Appendix A, will provide. Similarly, one may desire the number of searchers who viewed a certain number of results pages, addressed by SQL query #06, Appendix A.

One can calculate various statistical results on results page viewing, such as the average number of result pages viewed using SQL query #07, Appendix A.

An important aspect for system designers is results caching, for which one needs to know the number of repeat queries submitted by the entire set of searchers during the period. The SQL query #08, Appendix A, will tell us this information.

In order to understand how searchers are interacting with a search engine, the use of Boolean operators is an important feature. The SQL query #09, Appendix A, annotates a field

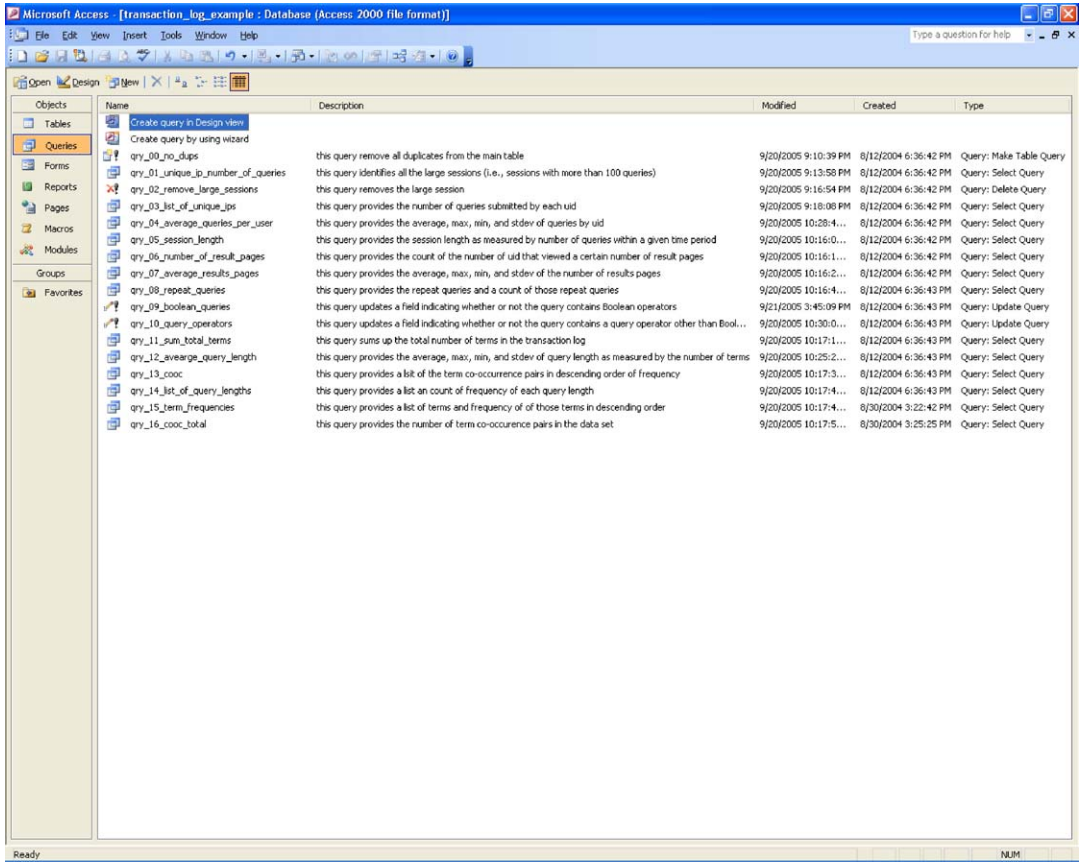


Fig. 4. Sequentially numbered and descriptively labeled queries for TLA.

identifying if there are Boolean operators within the queries. Since most search engines offer query syntax other than just Boolean operators, the SQL query #10, Appendix A, will annotate which queries contain this other query syntax.

The SQL query #11, Appendix A, provides a count of the number of terms within the transaction log. One certainly wants to know about query length; SQL query #12, Appendix A, provides statistics on query length. SQL query #13, Appendix A, provides the frequency of terms pairs within the transaction log. SQL query #14, Appendix A, provides a count of the occurrences of query lengths. SQL query #15, Appendix A, provides a count of the term frequencies. SQL query #26, Appendix A, provides a count of the frequency of term pairs within the transaction log.

The results from this series of queries both provides us a wealth of information about our data (e.g., occurrences of session lengths, occurrences of query length, occurrences of repeat queries, most used terms, most used term pairs), and serves as the basis for further investigations (e.g., session complexity, query structure, query modifications, term relationships).

### 3.6. An application for logging client-side actions

As a server-side data collection method, transaction logs typically do not contain the full range of user–system interactions. Therefore, researchers have to rely on other applications to capture records of these interactions.

Hancock-Beaulieu et al. (1990) supplemented their transaction logging with an application that included online questionnaires. Choo et al. (1998) had to develop their own logging software. Kelly (2004) used WinWhatWhere Investigator, which is a spy software package used to covertly “monitor” a person’s computer activities. Spy software has inherent disadvantages for use in user studies and evaluation, including granularity of data capture and privacy concerns. Toms, Freund, and Li (2004) developed the WiIRE system for conducting large scale evaluations. This system facilitates the evaluation of dispersed study participants; however, it is a server-side application focusing on the participant-interactions with Web server. As such, the entire “study” must occur within the WiIRE framework.

There are commercial applications for general-purpose (i.e., not specifically IR) user studies. An example is Morae 1.1 (<http://www.techsmith.com/products/morae/default.asp>) offered by TechSmith. Morae provides extremely detailed tracking of user actions, including video capture over a network. However, Morae is not specifically tailored for IR studies and captures so much information at such a fine granularity that it significantly complicates the data analysis process.

To assist in addressing this need, a software application was developed for use in conjunction with transaction log and other types of IR studies. The application is coded in a standard programming language (Visual Basic 6). It is easy to install and collects a wide range of user–systems interactions. The application logs much of the user interactions identified by prior research (Kelly & Teevan, 2003; Oard & Kim, 2001), along with the content of the interaction (i.e., URL, document, results listing, etc.). These implicit feedback actions and documents are referred to as action–object pairs (Jansen, 2003). We have validated the application in a series of user studies (cf. Jansen & Kroner, 2003; Jansen & McNeese, 2006) and have found the application to be extremely resilient, with near 100% operational effectiveness.

A description of the features and output of the application is presented, along with a uniform resource locator (URL) where interested researchers can download the application for use in their research projects and studies.

#### 3.6.1. Application description

The software application runs as an executable, generated from the Visual Basic programming environment. One can activate the application manually or via a bat file. The application has a Window’s interface for real time observation, which one can deactivate so that it does not display. The application logs interactions with the IR system, along with other applications, using Dynamic Data Exchange (DDE). Output is to a text file, with a specifiable location and an automatically generated unique filename. Fig. 5 displays the Window’s



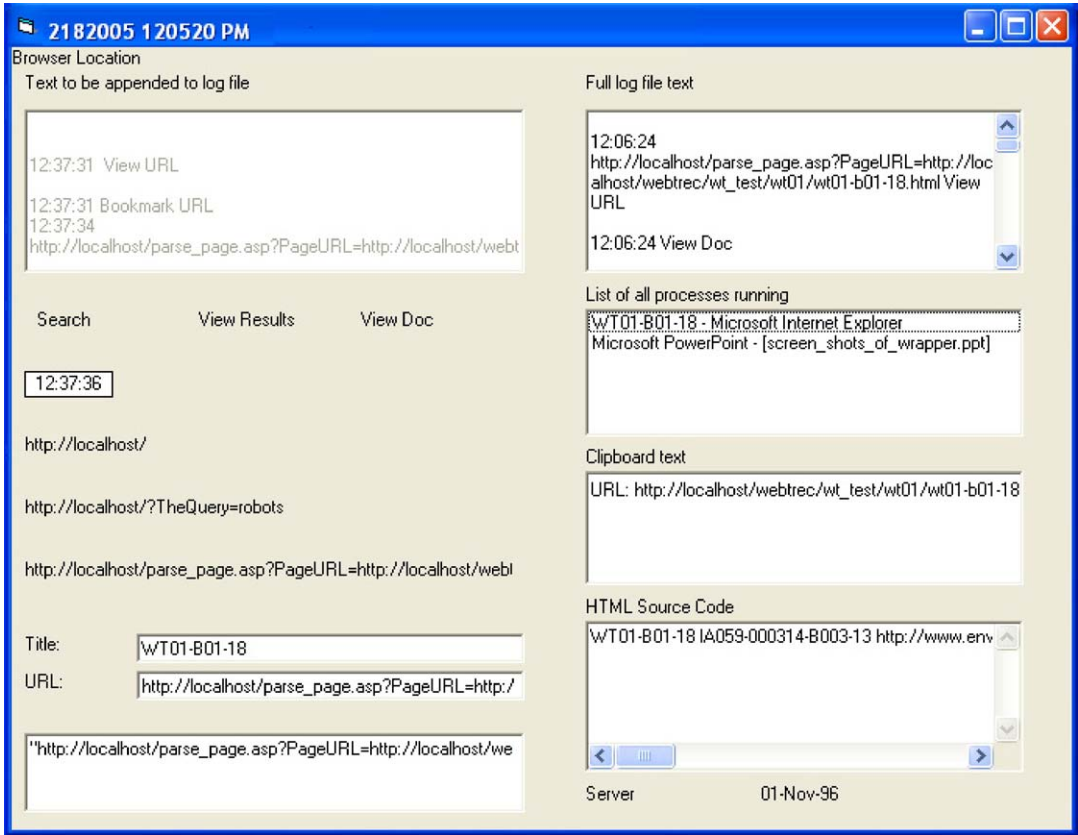


Fig. 5. The client-side application with action-object data displayed.

interface to the application. In the text file, each of the functional applications, shown in Fig. 5, is numbered as follows:

1. Log filename (generated automatically using date and time)
2. Running text of log file.
3. List of all processing running.
4. The current value of the clipboard.
5. HTML Source Code of the current page.
6. Text to be appended to log file.
7. The last three interactions logged.
8. Current system time.
9. Last three URLs visited.
10. Title and URL of current page.
11. Running list of URLs.

Table 1 shows an example of the application output (Table 3).

Table 3  
Transaction log of user interactions

Time stamp	Interaction
12:12:44	<a href="http://localhost/">http://localhost/</a>
12:12:44	Search RON (Back Space) BOTS
12:12:56	<a href="http://localhost/?TheQuery=robots">http://localhost/?TheQuery=robots</a> View URL
12:12:57	View Results
12:13:02	SCROLLED RESULTS
12:13:29	<a href="http://localhost/wt01/webtrec/wt01-b01-18.html">http://localhost/wt01/webtrec/wt01-b01-18.html</a>
12:13:30	View Doc
12:13:34	SCROLLED PAGE

In its current version, the application logs a wide range of user interactions, include interactions with the browser tool bar, interactions with the system clipboard, scrolling of results listing or documents, and numerous implicit feedback actions (Oard & Kim, 2001), such as bookmark, copy, print, save, and scroll.

#### 4. Discussion

It is certainly important to understand both the strengths and limitations of TLA for Web searching. First concerning the strengths, log analysis provides a method of collecting data from a great number of users. Given the current nature of the Web, transaction logs appear to be a reasonable and non-intrusive means of collecting user–system interaction data about the Web information-searching process from a large number of searchers. One can easily collect data on hundreds of thousands to millions of interactions, depending on the traffic of the Web site.

Second, one can collect this data inexpensively. The costs are the software and storage. Third, the data collection is unobtrusive, so the interactions represent the unaltered behavior of searchers. Finally, transactions log are, at present, the only method for obtaining significant amounts of data within the complex environment that is the Web (Dumais, 2002).

There are limitations of TLA, as with any methodology. First, there may be certain types of data not in the transaction log, individuals' identities being the most common example. An IP address typically represents the “user” in a transaction log. Since more than one person may use a computer, an IP address is an imprecise representation of the user. Search engines are overcoming this limitation somewhat by the use of cookies.

Second, there is no way to collect demographic data when using transaction logs in a naturalistic setting. This constraint is true of many non-intrusive naturalistic studies. However, there are several sources for demographic data on the Web population based on observational and survey data. From these data sources, one may get reasonable estimations of needed demographic data.

Third, a transaction log does not record the reasons for the search, the searcher motivations, or other qualitative aspects of use. This is certainly a limitation. In the instances where one needs this data, one should use TLA in conjunction with other data collection methods.

However, this invasiveness then intrudes on the unobtrusiveness, which is an inherent advantage of transaction logs as a data collection method.

Fourth, the logged data may not be complete due to caching of server data on the client machine or proxy servers. Although an often mentioned limitation, in reality, this is a minor concern for Web search engine research due to the method with which most search engines dynamically produce their results pages. For example, a user accesses the page of results from a search engine using the *Back* button of a browser. This navigation accesses the results page via the cache on the client machine. The Web server will not record this action. However, if the user clicks on any URL on that results page, functions coded on the results page redirects the click first to the Web server, from which the Web server records the visit to the Web site.

Following the literature review, we presented a three-step methodology for conducting TLA, namely, *collecting*, *preparing*, and *analyzing*. We then reviewed each step in detail, providing observations, guides, and lesson learned. The discussion focused on the organization at the ER level for the database, and we also presented the table design for standard search engine transaction logs and 16 queries one can use to conduct TLA. This methodology and detailed granularity serve as an excellent basis for novice or experienced transaction log researchers. Additionally, an actual transaction log file is provided with the manuscript as supplementary material.

Finally, an open source application for use during user studies of IR systems is presented. The application is focused on the typical interactions of searchers, thereby providing the needed granularity of data for fruitful analysis, without logging overwhelming amounts of data that slow the data analysis process. The application is currently available for download at <http://ist.psu.edu/faculty/jansen/>. In future research on this application, we aim to increase the number of user interactions logged.

## 5. Conclusion

Transaction logs are powerful tools for collecting data on the interactions between users and systems. Using this data, TLA can provide significant insights into user–system interactions, and it complements other methods of analysis by overcoming the limitations inherent in these methods. With respect to shortcomings, one can combine TLA with other data collection methods or other research results to improve the robustness of the analysis. Overall, TLA is a powerful tool for Web-searching research, and the TLA process outlined here can be helpful in future Web-searching research endeavors.

## Appendix A

SQL Query 00:

```
qry_00_no_dups  
SELECT tbl_main.uid, tbl_main.date, tbl_main.search_url, Count(tbl_main.search_url)
```

```

AS CountOfsearch_url, First(tbl_main.thetime) AS FirstOfthetime,
  First(tbl_main.qid) AS FirstOfqid INTO tbl_searching_episodes
FROM tbl_main
GROUP BY tbl_main.uid, tbl_main.date, tbl_main.search_url;

```

SQL Query 01:

```

qry_01_unique_ip_number_of_queries
SELECT tbl_searching_episodes.uid
FROM tbl_searching_episodes
GROUP BY tbl_earching_episodes.uid
HAVING (((Count(tbl_searching_episodes.uid))>=100));

```

SQL Query 02:

```

qry_02_remove_large_sessions
DELETE tbl_searching_episodes.qid, tbl_searching_episodes.uid, tbl_searching_
episodes.thetime, tbl_searching_episodes.search_url,
tbl_searching_episodes.qtot, tbl_searching_episodes.uid
FROM tbl_searching_episodes
WHERE (((tbl_searching_episodes.uid)="[inset values here]"));

```

SQL Query 03:

```

qry_03_list_of_unique_ips
SELECT tbl_searching_episodes.uid, Count(tbl_searching_episodes.search_url)
AS CountOfsearch_url
FROM tbl_searching_episodes
GROUP BY tbl_searching_episodes.uid
ORDER BY Count(tbl_searching_episodes.search_url) DESC;

```

SQL Query 04:

```

qry_04_average_queries_per_user
SELECT Avg(qry_03_list_of_unique_ips.CountOfsearch_url) AS AvgOfCount
Ofsearch_url
FROM qry_03_list_of_unique_ips;

```

SQL Query 05:

```

qry_05_session_length
SELECT qry_03_list_of_unique_ips.CountOfsearch_url, Count(qry_03_list_of_
unique_ips.CountOfsearch_url) AS CountOfCountOfsearch_url

```

```

FROM qry_03_list_of_unique_ips
GROUP BY qry_03_list_of_unique_ips.CountOfsearch_url
ORDER BY Count(qry_03_list_of_unique_ips.CountOfsearch_url)
DESC;

```

SQL Query 06:

```

qry_06_number_of_result_pages
SELECT tbl_searching_episodes.qtot,
Count(tbl_searching_episodes.qtot) AS CountOfqtot
FROM tbl_searching_episodes
GROUP BY tbl_searching_episodes.qtot
ORDER BY tbl_searching_episodes.qtot;

```

SQL Query 07:

```

qry_07_average_results_pages
SELECT Avg(tbl_searching_episodes.qtot) AS AvgOfqtot
FROM tbl_searching_episodes;

```

SQL Query 08:

```

qry_08_repeat_queries
SELECT tbl_searching_episodes.search_url, Count(tbl_searching_episodes.
search_url) AS CountOfsearch_url
FROM BY tbl_searching_episodes.search_url
ORDER BY Count(tbl_searching_episodes.search_url) DESC;

```

SQL Query 09:

```

qry_09_boolean_queries
UPDATE tbl_searching_episodes SET tbl_searching_episodes.boolean = True
WHERE (((tbl_searching_episodes.search_url) Like "* and *" Or
(tbl_searching_episodes.search_url) Like "* or *" Or
(tbl_searching_episodes.search_url) Like "* and not *"));

```

SQL Query 10:

```

qry_10_query_operators
UPDATE tbl_searching_episodes SET tbl_searching_episodes.operator = True
WHERE (((tbl_searching_episodes.search_url) Like "*" Or (tbl_searching_episodes.
search_url) Like "+*" Or (tbl_searching_episodes.search_url) Like "[*]" Or
(tbl_searching_episodes.search_url) Like "[?]*"));

```

## SQL Query 11:

```

qry_11_sum_total_terms
SELECT Sum(tblterms.tfreq) AS SumOftfreq
FROM tblterms;

```

## SQL Query 12:

```

qry_12_average_query_length
SELECT Avg(tbl_searching_episodes.qry_length) AS AvgOfqry_length
FROM tbl_searching_episodes;

```

## SQL Query 13:

```

qry_13_cooc
SELECT tblterms.term, tblterms.term, tblcooc.tot
FROM tblterms INNER JOIN tblcooc ON (tblterms.termid = tblcooc.cid2)
AND (tblterms.termid = tblcooc.cid1)
ORDER BY tblcooc.tot DESC;

```

## SQL Query 14:

```

qry_14_list_of_query_lengths
SELECT tbl_searching_episodes.qry_length,
Count(tbl_searching_episodes.qry_length) AS CountOfqry_length
FROM tbl_searching_episodes
GROUP BY tbl_searching_episodes.qry_length
ORDER BY Count(tbl_searching_episodes.qry_length) DESC;

```

## SQL Query 15:

```

qry_15_term_frequencies
SELECT tblterms.tfreq
FROM tblterms
GROUP BY tblterms.tfreq
ORDER BY tblterms.tfreq;

```

## SQL Query 16:

```

qry_16_cooc_total
SELECT Sum(tblcooc.tot) AS SumOfTot
FROM tblcooc;

```

## Appendix B

### Queries ordered by use with descriptions

Query title	Query description
qry_00_no_dups	This query remove all duplicates from the main table
qry_01_unique_ip_number_of_queries	This query identifies all the large sessions (i.e., sessions with more than 100 queries)
Query title	Query description
qry_02_remove_large_sessions	This query removes the large session
qry_03_list_of_unique_ips	This query provides the number of queries submitted by each uid
qry_04_average_queries_per_user	This query provides the average, max, min, and stdev of queries by uid
qry_05_session_length	This query provides the session length as measured by number of queries within a given time period
qry_06_number_of_result_pages	This query provides the count of the number of uid that viewed a certain number of result pages
qry_07_average_results_pages	This query provides the average, max, min, and stdev of the number of results pages
qry_08_repeat_queries	This query provides the repeat queries and a count of those repeat queries
qry_09_boolean_queries	This query updates a field indicating whether or not the query contains Boolean operators
qry_10_query_operators	This query updates a field indicating whether or not the query contains a query operator other than Boolean
qry_11_sum_total_terms	This query sums up the total number of terms in the transaction log
qry_12_avearge_query_length	This query provides the average, max, min, and stdev of query length as measured by the number of terms
qry_13_cooc	This query provides a list of the term co-occurrence pairs in descending order of frequency
qry_14_list_of_query_lengths	This query provides a list an count of frequency of each query length
qry_15_term_frequencies	This query provides a list of terms and frequency of those terms in descending order
qry_16_cooc_total	This query provides the number of term co-occurrence pairs in the data set

## Appendix C. Supplementary data

The supplemental file is a transaction log from the Excite search engine containing over one million records. Supplementary data (search engine transaction log) associated with this article can be found, in the online version, at [doi:10.1016/j.lisr.2006.06.005](https://doi.org/10.1016/j.lisr.2006.06.005).

## References

- Baeza-Yates, R., & Castillo, C. (2001). *Relating Web structure and user search behavior*. Paper presented at the 10th World Wide Web Conference, Hong Kong, China May.
- Bains, S. (1997). End-user searching behavior: Considering methodologies. *The Katharine Sharp Review, 1(4)*. Retrieved June 27, 2006, from <http://mirrored.ukoln.ac.uk/lis-journals/review/review/winter1997/bains.html>

- Bates, M. J. (1990). Where should the person stop and the information search interface start? *Information Processing and Management*, 26, 575–591.
- Belkin, N., Cool, C., Stein, A., & Theil, S. (1995). Cases, scripts, and information-seeking strategies: On the design of interactive information retrieval systems. *Expert Systems With Applications*, 9, 379–395.
- Blecic, D., Bangalore, N. S., Dorsch, J. L., Henderson, C. L., Koenig, M. H., & Weller, A. C. (1998). Using transaction log analysis to improve OPAC retrieval results. *College and Research Libraries*, 59(1), 39–50.
- Borgman, C. L., Hirsh, S. G., & Hiller, J. (1996). Rethinking online monitoring methods for information retrieval systems: From search product to search process. *Journal of the American Society for Information Science*, 47, 568–583.
- Cacheda, F., & Viña, Á. (2001). *Experiences retrieving information in the World Wide Web*. Paper presented at the 6th IEEE Symposium on Computers and Communications, Hammamet, Tunisia July.
- Chamberlain, K. (1995, November). *What is Grounded Theory?* Retrieved September 17, 2005, from <http://kerlins.net/bobbi/research/qualresearch/bibliography/gt.html>
- Chau, M., Fang, X., and Sheng, O.R.L. (In press). Analyzing the query logs of a Web site search engine. *Journal of the American Society for Information Science and Technology*.
- Choo, C., & Turnbull, D. (2000). Information seeking on the Web: An integrated model of browsing and searching. *First Monday*, 5(2). Retrieved June 27, 2006, from [http://firstmonday.org/issues/issue5\\_2/choo/index.html](http://firstmonday.org/issues/issue5_2/choo/index.html)
- Choo, C., Betlor, B., & Turnbull, D. (1998). *A behavioral model of information seeking on the Web: Preliminary results of a study of how managers and IT specialists use the Web*. Paper presented at the 61st Annual Meeting of the American Society for Information Science, Pittsburgh, PA.
- Cooper, M. D. (1998). Design considerations in instrumenting and monitoring Web-based information retrieval systems. *Journal of the American Society for Information Science*, 49, 903–919.
- Croft, W., Cook, R., & Wilder, D. (1995, June). *Providing government information on the Internet: Experiences with THOMAS*. Paper presented at the Digital Libraries Conference, Austin, TX.
- Drott, M. C. (1998). *Using Web server logs to improve site design*. Paper presented at the 16th Annual International Conference on Computer Documentation, Quebec, Canada.
- Dumais, S.T. (2002, May). *Web experiments and test collections*. Retrieved April 20, 2003, from <http://www2002.org/presentations/dumais.pdf>
- Efthimiadis, E. N., & Robertson, S. E. (1989). Feedback and interaction in information retrieval. In C. Oppenheim (Ed.), *Perspectives in information management* (pp. 257–272). London: Butterworths.
- Fourie, I. (2002, October). A review of Web information-seeking/searching studies (2000–2002): Implications for research in the South African context. Paper presented at Progress in Library and Information Science in Southern Africa: 2d Biannual DISSAnet Conference, Pretoria, South Africa.
- Fourie, I., & van den Berg, H. (2003, June). *A story told by Nexus transaction logs: What to make of it*. Paper presented at 7th Southern African Online Meeting, Muldersdrift, South Africa.
- Glaser, B., & Strauss, A. (1967, June). *The discovery of grounded theory: Strategies for qualitative research*. Chicago: Aldine Publishing Co.
- Griffiths, J.R., Hartley, R.J., & Willson, J.P. (2002). An improved method of studying user–system interaction by combining transaction log analysis and protocol analysis. *Information Research*, 7(4). Retrieved June 27, 2006, from <http://InformationR.net/ir/7-4/paper139.html>
- Hancock-Beaulieu, M. (2000). Interaction in information searching and retrieval. *Journal of Documentation*, 56, 431–439.
- Hancock-Beaulieu, M., Robertson, S., & Nielsen, C. (1990). *Evaluation of online catalogues: An assessment of methods (BL Research Paper 78)*. London: The British Library Research and Development Department.
- Hargittai, E. (2002). Beyond logs and surveys: In-depth measures of people's Web use skills. *Journal of the American Society for Information Science and Technology*, 53, 1239–1244.
- He, D., Göker, A., & Harper, D. J. (2002). Combining evidence for automatic Web session identification. *Information Processing and Management*, 38, 727–742.
- Jansen, B. J. (2003, October). Designing automated help using searcher system dialogues. Paper presented at the 2003 IEEE International Conference on Systems, Man and Cybernetics, Washington, D.C., USA.



- Jansen, B. J., & Kroner, G. (2003, April). *The impact of automated assistance on the information retrieval process*. Paper presented at The ACM CHI 2003 Conference on Human Factors in Computing Systems, Fort Lauderdale, Florida.
- Jansen, B. J., & McNeese, M. D. (2006). Evaluating the effectiveness of and patterns of Interactions with automated searching assistance. *Journal of the American Society of Information Science and Technology*, 56(14), 1480–1503.
- Jansen, B. J., & Pooch, U. (2001). Web user studies: A review and framework for future work. *Journal of the American Society of Information Science and Technology*, 52, 235–246.
- Jansen, B. J., & Spink, A. (2003, June). *An analysis of Web information seeking and use: Documents retrieved versus documents viewed*. Paper presented at 4th International Conference on Internet Computing, Las Vegas, Nevada.
- Jansen, B. J., & Spink, A. (2005). How are we searching the World Wide Web? A comparison of nine search engine transaction logs. *Information Processing and Management*, 42, 248–263.
- Jansen, B. J., Spink, A., & Saracevic, T. (2000). Real life, real users, and real needs: A study and analysis of user queries on the Web. *Information Processing and Management*, 36, 207–227.
- Jansen, B. J., Spink, A., & Pederson, J. (2005). Trend analysis of AltaVista Web searching. *Journal of the American Society for Information Science and Technology*, 56, 559–570.
- Jones, S., Cunningham, S., & McNab, R. (1998, June). *Usage analysis of a digital library*. Paper presented at the Third ACM Conference on Digital Libraries, Pittsburgh, PA.
- Kelly, D. (2004). *Understanding implicit feedback and document preference: A naturalistic user study*. Rutgers, The State University of New Jersey: New Brunswick.
- Kelly, D., & Teevan, J. (2003). Implicit feedback for inferring user preference: A bibliography. *SIGIR Forum*, 37(2), 18–28.
- Kinsella, J., & Bryant, P. (1987). Online public access catalogue research in the United Kingdom: An overview. *Library Trends*, 35, 619–629.
- Korfhage, R. (1997). *Information storage and retrieval*. New York: Wiley.
- Kurth, M. (1993). The limits and limitations of transaction log analysis. *Library Hi Tech.*, 11(2), 98–104.
- Lin, S.-J. (2002, August). *Design space of personalized indexing: Enhancing successive Web searching for transmuting information problems*. Paper presented at the Eighth Americas Conference on Information Systems, Dallas, Texas.
- Meister, D., & Sullivan, D. (1967). *Evaluation of user reactions to a prototype on-line information retrieval system: Report to NASA by the Bunker-Ramo Corporation. Report Number NASA CR-918*. Oak Brook, IL: Bunker-Ramo Corporation.
- Millsap, L., & Ferl, T. (1993). Search patterns of remote users: An analysis of OPAC transaction logs. *Information Technology and Libraries*, 11, 321–343.
- Miwa, M. (2001, February). *User situations and multiple levels of users goals in information problem solving processes of AskERIC users*. Paper presented at the 2001 Annual Meeting of the American Society for Information Sciences and Technology, San Francisco, CA, USA.
- Montgomery, A., & Faloutsos, C. (2001). Identifying Web browsing trends and patterns. *IEEE Computer*, 34(7), 94–95.
- Moukdad, H., & Large, A. (2001). Users' perceptions of the Web as revealed by transaction log analysis. *Online Information Review*, 25, 349–358.
- Nicholas, D., Huntington, P., Lievesley, N., & Withey, R. (1999). Cracking the code: Web log analysis. *Online and CD ROM Review*, 23, 263–269.
- Oard, D., & Kim, J. (2001, October–November). *Modeling information content using observable behavior*. Paper presented at the 64th Annual Meeting of the American Society for Information Science and Technology, Washington, D.C., USA.
- Ozmutlu, S., Ozmutlu, H. C., & Spink, A. (2003, June). *A study of multitasking Web searching*. Paper presented at the IEEE ITCC'03: International Conference on Information Technology: Coding and Computing, Las Vegas, Nevada.

- Park, S., Bae, H., & Lee, J. (2005). End user searching: A Web log analysis of NAVER, a Korean Web search engine. *Library and Information Science Research*, 27, 203–221.
- Penniman, W. D. (1975, October). *A stochastic process analysis of online user behavior*. Paper presented at the Annual Meeting of the American Society for Information Science, Washington, DC.
- Peters, T. (1993). The history and development of transaction log analysis. *Library Hi Tech.*, 42(11), 41–66.
- Phippen, A., Sheppard, L., & Furnell, S. (2004). A practical evaluation of Web analytics. *Internet Research: Electronic Networking Applications and Policy*, 14, 284–293.
- Pitkow, J. E. (1997, April). *In search of reliable usage data on the WWW*. Paper presented at Santa Clara, CA, the Sixth International World Wide Web Conference.
- Rice, R. E., & Borgman, C. L. (1983). The use of computer-monitored data in information science. *Journal of the American Society for Information Science*, 44, 247–256.
- Ross, N., & Wolfram, D. (2000). End user searching on the Internet: An analysis of term pair topics submitted to the excite search engine. *Journal of the American Society for Information Science*, 51, 949–958.
- Sandore, B., Flaherty, P., & Kaske, N. K. (1993). A manifesto regarding the future of transaction log analysis. *Library Hi Tech.*, 11(2), 105–111.
- Saracevic, T. (1997, November). *Extension and application of the stratified model of information retrieval interaction*. Paper presented at the Annual Meeting of the American Society for Information Science, Washington, DC.
- Silverstein, C., Henzinger, M., Marais, H., & Moricz, M. (1999). Analysis of a very large Web search engine query log. *SIGIR Forum*, 33(1), 6–12.
- Spink, A. (2004). Multitasking information behavior and information task switching: An exploratory study. *Journal of Documentation*, 60, 336–345.
- Spink, A., & Jansen, B. J. (2004). *Web search: Public searching of the Web*. New York: Kluwer.
- Spink, A., Wilson, T., Ellis, D., & Ford, F. (1998). Modeling users' successive searches in digital environments. *D-Lib Magazine*. Retrieved June 27, 2006, from <http://www.dlib.org/dlib/april98/04spink.html>
- Sullivan, D. (2001). *SpiderSpotting: When a search engine, robot or crawler visits*. Retrieved August 5, 2003, from <http://www.searchenginewatch.com/webmasters/article.php/2168001>
- Toms, E. G., Freund, L., & Li, C. (2004). WiIRE: The Web interactive information retrieval experimentation system prototype. *Information Processing and Management*, 40, 655–675.
- Wang, P., Berry, M., & Yang, Y. (2003). Mining longitudinal Web queries: Trends and patterns. *Journal of the American Society for Information Science and Technology*, 54, 743–758.
- Wu, K.-L., Yu, P. S., & Ballman, A. (1998). SpeedTracer: A Web usage mining and analysis tool. *IBM Systems Journal*, 37(1), 89–107.
- Yuan, W., & Meadow, C. T. (1999). A study of the use of variables in information retrieval user studies. *Journal of the American Society for Information Science*, 50, 140–150.