**M. Hirose (Ed.),** *INTERACT '01: Proceedings of the IFIP TC 13 International Conference on Human-Computer Interaction* (Tokyo, Japan, July 9-13, 2001), pp. 537-544. IOS Press, Amsterdam.

Preprint version

# How Are Searching and Reading Intertwined during Retrieval from Hierarchically Structured Documents?

**Morten Hertzum[1], Mounia Lalmas[2] & Erik Frøkjær[3]**

[1] Centre for Human-Machine Interaction, Risø National Laboratory, Denmark
[2] Department of Computer Science, Queen Mary, University of London, UK
[3] Datalogi, Roskilde Universitetscenter, Denmark

morten.hertzum@risoe.dk, mounia@dcs.qmw.ac.uk, erikf@ruc.dk

**Abstract**: Effective use of information retrieval systems requires that users know when to – temporarily – cease searching to do some reading and where to start reading. In hierarchically structured documents, users can to some extent interchange searching and reading by entering the text at different levels in the structure. Based on an experiment where 83 subjects solved 20 tasks each, we find that the subjects spend at least one third of their time reading, irrespective of whether the system they use offers browsing and/or querying. The subjects prefer reading the text in chunks that span multiple levels of the structure. As much as 21% of the tasks are solved by subjects who enter the text at a level above the texts containing the answer and rely on reading from there. In relation to queries, multi-level texts are used to extend hits with more detail or, occasionally, more context. Designers should consider how information retrieval systems could exploit document structure to return the best points to support reading, rather than merely hits.

**Keywords:** Information retrieval, structured documents, searching, reading, context information, user study

## 1 Introduction

In designing user interfaces for information retrieval systems we need an understanding of the information-seeking behaviour of the user population. Whereas interfaces for querying rely on the users to provide an initial specification of what they want before they are presented with the first examples of documents, interfaces for browsing aim to capitalise on the greater ability of humans to recognise what is wanted over being able to describe it. This exemplifies inherently different assumptions about how users intertwine the realisation/ formulation of what they want (searching) and the exploration/comprehension of what is available (reading). In this study we analyse how users intertwine searching and reading during retrieval from hierarchically structured documents. This is done by analysing data from the TeSS experiment (Frøkjær et al, 2000; Hertzum & Frøkjær, 1996) where 83 subjects solved a number of information retrieval tasks using four different modes of the TeSS system.

Information about the structure of the documents in text collections has been used to support retrieval in manifold ways. In query evaluation algorithms, words from titles or abstracts are for example often considered more precise descriptors of document contents than words from anywhere in the documents (for example, Cleveland et al, 1984; Tenopir & Ro, 1990). In presenting the results from queries, hits are often grouped according to their position in the structure. For example, many search engines on the World Wide Web group the hits by the web site they belong to. Also, information about the position of the hits in the structure is sometimes presented along with the hits to help the user in assessing them (for example, Hearst, 1995; Robertson et al, 1993). Finally, in enabling effective browsing of large documents the structure of the documents can be used to define fisheye views that balance the need for local detail against the need for global context (Furnas, 1981/1999).

In the TeSS experiment, which comprises both browsing and querying, the hierarchical structure of the text collection forms the basis for the browsing facilities, the presentation of query results, and the extent of the chunks of text that users bring up for reading. The users' information-seeking behaviour is affected by and makes use of the hierarchical structure of the text collection, as it is reflected in the retrieval modes. We will analyse this interaction between information-seeking behaviour and hierarchically structured text by means of a distinction between searching and reading:

- *Searching* encompasses the activities that go into finding texts that are perceived as sufficiently relevant to the task at hand to warrant closer inspection. In the TeSS experiment searching consists of querying by means of full-text, Boolean queries and browsing by means of expanding and collapsing the table of contents of the text collection.
- *Reading* encompasses looking at the actual texts, irrespective of whether it is done to carefully scrutinise the contents or to skim through it. In the TeSS experiment reading is done in separate text viewers opened from the main window where the searching takes place.

Effective use of information retrieval systems requires that the users learn when to – temporarily – cease searching to do some reading and where to start reading. To support users in making these decisions, designers need an understanding of what constitutes the best entry points into the text collection. In hierarchically structured texts the entry

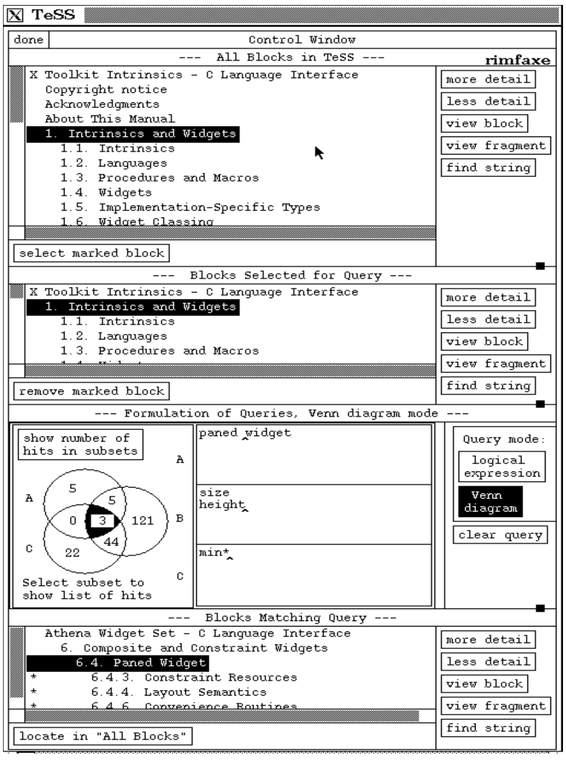


**Figure 1:** The TeSS control window consists of four subwindows: (1) the table of contents, (2) the selection list, (3) the query subwindow (currently in Venn diagram mode), and (4) the hit list in which hits are marked with asterisks.

points can be set at different levels relative to the information that is deemed relevant (Lalmas & Moutogianni, 2000). For example, access to a whole section should be provided when all – or most – of its subsections contain information that is deemed relevant. To investigate what may constitute a best entry point, we compare the texts users actually opened for reading (the preferred entry points) with both the hits returned from queries and the texts that contain the answers to the tasks. Finally, we draw some conclusions of our study regarding the design of systems for accessing hierarchically structured documents.

# 2 The TeSS System

Access to the documentation necessary to solve the experimental tasks was provided through an information retrieval system, called TeSS, designed for programmers with a need for consulting documentation while coding. The user interface of TeSS includes a control window and any number of text viewers. The control window (see Figure 1) is designed to display, at all times, all functions available to the user and all information defining the current state of the search. TeSS gives online access to 3 MB of documentation relevant when developing graphical user interfaces in the X Window System (the C language interface manuals for Xlib, X Toolkit Intrinsics, and Athena Widget Set).

TeSS can be operated in four different modes, each providing the user with a different set of retrieval facilities:

- BROWSE. In TeSS, browsing can be done by expanding and collapsing entries in the table of contents and by searching the table of contents for specific strings. The text itself is presented in separate windows.
- LOGICAL. A mode of TeSS offering conventional Boolean retrieval where queries are logical expressions built of query terms, ANDs, ORs, NOTs, parentheses, and wildcards.
- VENN. In this mode of TeSS queries are expressed by means of a Venn diagram which replaces Boolean operators with a, supposedly, more immediately understandable graphical image of intersecting sets.
- ALL. The whole of TeSS offering the combination of BROWSE, LOGICAL, and VENN.

In TeSS, browsing is limited to the table of contents and, thus, emphasises the structure of the documentation. Similarly, the results of queries are presented in context by preceding them with information about the manual, chapter, subchapter etc. in which the hits are located (see Figure 1, subwindow 4). The hits in the hit list are in the same format as the entries in the table of contents; i.e., they consist of the section number followed by the heading text. Access to the actual text is provided through text viewers, which can be opened at any level in the text hierarchy. *View block* opens a text viewer with the whole subhierarchy of text belonging to the selected heading (see Figure 2). This way the user can retrieve, for example, an entire chapter for closer inspection. *View fragment*, on the other hand, opens a text viewer that presents the text belonging to a heading with exclusion of text belonging to its subheadings (see Figure 2). This may be useful when inspecting hits from a query because the presented text is reduced to the smallest part that matches the query.

Prior to the experiment TeSS was tested through a couple of informal think-aloud sessions. After an initial overwhelming impression, no one seemed to have problems understanding and using the system. Note, however, that opening a text viewer is a rather slow operation, taking 5-15 seconds.

# 3 The TeSS Experiment

The purpose of the TeSS experiment was to compare browsing and different forms of querying with respect to usage effectiveness and the subjects' information-seeking behaviour.

## 3.1 Subjects

The subjects were 87 students in their third year of a bachelor degree in computer science. The project was a mandatory part of the students' education but participation in the experiment by allowing the data collection to take place was voluntary and anonymous. The subjects were first-time users of TeSS and had no prior knowledge of the programming tools on which the tasks were based.

## 3.2 Tasks

In the TeSS experiment each subject solved 20 information retrieval tasks concerning whether and how certain interface properties could be achieved in a graphical X Window System interface. As preparation, the subjects completed two practice tasks. To answer the 20 information retrieval tasks

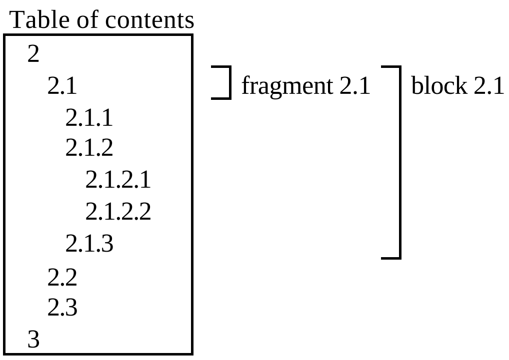


**Figure 2:** Text fragments and text blocks.

| Grade | Description |
|---|---|
| 1 | Failure, a completely wrong answer |
| 2 | Inadequate or partially wrong answer |
| 3 | Reasonable but incomplete answer |
| 4 | Good and adequate answer |
| 5 | Brilliant answer |

**Table 1**: The five-point scale used to grade the tasks.

the subjects had to identify the relevant user interface objects, e.g. widgets, methods, and resources, and outline an implementation. As the subjects were unfamiliar with the X Window System, the tasks involved a substantial element of learning in addition to the need for retrieving specific pieces of information. Some tasks were formulated in the context of the X Window System in general; others took the user interface of TeSS as their point of departure. Two examples of tasks used in the experiment are:

*Task 5*. Radio buttons are used in situations where exactly one option must be chosen from a group of options. Which widget class is used to implement radio buttons?

*Task 11*. The caption on the button "done" should be changed to "quit". How is that done?

The quality of the subjects' solutions of the tasks was assessed by the first author and expressed by a grade on a five-point scale, see Table 1. As an example, medium and high quality solutions of task 5 must identify toggle widgets as the relevant widget class. A brilliant answer also explains the use of radio groups to cluster the toggle widgets.

## 3.3 Procedure

The experiment was explained to the subjects at a lecture, after which the subjects had ten days to complete the tasks. The subjects received a manual for TeSS and a two-page walk-up-and-use introduction. The system itself was available on terminals to which students have access 24 hours a day. The experiment employed a within-groups design where all subjects solved the tasks in the same sequence and each subject was required to use all retrieval modes. To avoid order effects, the subjects were exposed to the retrieval modes in a systematically varied order. The 20 information retrieval tasks were clustered into five blocks (one for each of the four modes of TeSS plus a manual mode where the subjects used hardcopies of the documentation). The first block was solved with one of the five retrieval modes, the second block with one of the remaining four retrieval modes. Thus the permutations of the modes on the two first blocks divided the subjects into 20 groups. The number of subjects did not allow all 5! sequences of the five modes to be included, and the 20 groups were not divided further. Rather, the order of the three remaining modes was kept the same within each group.

The data collected from the experiment include a detailed log of the subjects' interaction with TeSS. The interaction log gives a time-stamped account of the commands executed by the subjects. It also includes task demarcation and solutions reached, both obtained from a subsystem administering the subjects' access to TeSS. This task-handling module makes it possible to let the subjects work unsupervised while at the same time enforcing a strict experimental procedure. The task-handling module presents the tasks to the subject one at a time, gives access to the retrieval mode to be used by that subject when solving that particular task, and records his or her solution.

### 3.4 Data Analysis

The 87 subjects received 20 information retrieval tasks each, giving a potential total of 1740 solutions. However, 113 solutions were not submitted; 19 were excluded because they included a more than one hour long period with no logged user activity; 17 were excluded due to technical problems with TeSS; 14 were excluded because it was impossible to judge the quality of the solution; and 2 were excluded because they were solved poorly in less than two minutes, i.e., without any attempt to solve the task. Finally, 4 subjects were excluded because they clearly did not take the experiment seriously. Thus, 11% of the solutions were not submitted or excluded. Of the remaining 1555 solutions 1236 were solved using one of the four modes of TeSS. The data from the use of hardcopies of the documentation are too coarse-grained to be included in the present study. The 1236 solutions analysed in this study are the result of 565 hours of work performed by 83 subjects.

For the purpose of this paper a number of overall measures of the subjects' information-seeking behaviour have been extracted from the interaction log, for example the percentage of the task completion time spent reading as opposed to searching. The data analysis should ensure that these behavioural measures reflect differences between the retrieval modes only. Due to the within-groups design, differences between subjects have to be accounted for; otherwise the dispersion of the subjects' mean behaviour, for example between consistently searching-biased and consistently reading-biased subjects, might blur differences pertaining to the retrieval modes. Following Dix et al (1998, pp. 423-427) this is avoided by aligning each subject's behaviour with the global mean behaviour. Differences between the tasks must also be taken into account because not all tasks have been solved exactly equally many times with each retrieval mode. For each behavioural measure, v, $subject_i$'s behaviour on $task_j$ is adjusted as follows:

$$v'_{ij} = v_{ij} - (ms_i - m) - (mt_j - m)$$

where $v'_{ij}$ is the adjusted behaviour, $v_{ij}$ is the measured behaviour, $(ms_i - m)$ is the difference between the subject mean and the global mean, and $(mt_j - m)$ is the difference between the task mean and the global mean. After this adjustment, the behavioural measures have been analysed by pairwise comparisons of the retrieval modes using t-tests.

Finally, the text fragments containing the answers to the tasks were identified (by the first author) to enable analysis of the relationship between these target texts and the texts inspected by the subjects. The 20 tasks had an average of 2.7 target texts. For 15 of the tasks each target text contained the information needed to solve the task; the remaining five tasks required the combination of information from several target texts.

## 4 Results and Discussion

Hertzum & Frøkjær (1996) found that the four modes of TeSS are equal in terms of the quality of the subjects' solutions of the tasks but differ with respect to the time spent reaching the solutions (see the first row in Table 2). Below we will investigate how the subjects combine searching and reading, and to what extent this aspect of their information-seeking behaviour depends on the retrieval mode they have available.

### 4.1 Behavioural Measures

Table 2 summarises how reading entered into solving the tasks. The subjects use *View block* the least in BROWSE and the most in LOGICAL. The average difference between these two retrieval modes is one *View block* command per task – the only significant difference for this behavioural measure. For *View fragment*, the only significant difference is the 51% increase going from VENN to ALL. The total number of viewers opened with either *View block* or *View fragment* does not differ significantly across the four retrieval modes (Hertzum & Frøkjær, 1996). However, the subjects display a notable preference for *View block*, which is used more than twice as often as *View fragment*. This preference is significant for each of the four retrieval modes (t-tests, $p<0.0001$).

For all four retrieval modes, the first 25-28% of the task completion time is spent searching and only then do the subjects open their first text viewer to start reading. The subjects stop reading well before the end of a task and spend the last 17-22% of their task completion time doing additional searching and writing down their answer to the task.

| | BROWSE | LOGICAL | VENN | ALL | Significant differences (t-tests, p<0.05) |
|---|---|---|---|---|---|
| Time spent on task (decimal minutes) | 23.08 | 30.02 | 25.53 | 30.77 | BROWSE < VENN < LOGICAL, ALL |
| *View block* (commands per task) | 3.56 | 4.67 | 4.10 | 4.05 | BROWSE < LOGICAL |
| *View fragment* (commands per task) | 1.47 | 1.68 | 1.32 | 1.99 | VENN < ALL |
| Time to first viewer (pct. of time spent on task) | 25% | 25% | 28% | 28% | LOGICAL < ALL, VENN<br>BROWSE < VENN |
| Time from last viewer (pct. of time spent on task) | 22% | 19% | 17% | 18% | VENN, ALL, LOGICAL < BROWSE |
| Time in text * (pct. of time spent on task) | 39% | 38% | 35% | 35% | VENN, ALL < LOGICAL, BROWSE |

* Time in text is defined as the time from a viewer is opened to the following command is issued, accumulated over all viewers. As some subjects left viewers open for later reference, the time in text gives a lower bound for the reading time.

**Table 2:** Behavioural measures.

Accumulated over all text viewers opened during a task, the time spent reading, as opposed to searching, amounts to 35-39% of the task completion time. The time spent reading the text in a viewer is defined as the time from the viewer is opened to the following command is issued. For 64% of the 7058 viewers this command closes the viewer, suggesting that the subjects feel they derived everything useful from the text during their first reading of it. The remaining 36% of the viewers stay open when the subjects resume searching and they may have returned to these viewers later. Thus, the time in text gives a lower bound for the reading time: The average subject spend *at least* one third of his or her time reading, irrespective of retrieval mode. In the middle part of the tasks (the period from the first to the last viewer) the subjects spend more time reading than searching.

## 4.2 Preferred Entry Points versus Hits

In TeSS, queries are matched against text fragments and, hence, a hit corresponds to the text belonging to a heading with exclusion of text belonging to its subheadings. That is, the hits are as specific as possible (this is in accordance with the suggestion advanced in Chiaramella et al, 1996). The users are however not restricted to this maximum level of specificity in their access to the texts. Apart from opening hits with *View fragment*, the users can: (1) Get more detail by opening hits with *View block*. This gives the user access to a text block that starts with the text fragment that matches the query but also includes the text belonging to its subheadings. (2) Get more context by opening one of the headings that lead in to the hits (see Figure 3).

In situations – such as the TeSS experiment – where solving a task involves a substantial element of learning we must expect that people will often need more information than they can find in hits that are as specific as possible. The lead-in headings were included in the hit list to position the hits in the overall structure of the text collection and to give the users access to all the higher-level text blocks that contain hits. We saw above (Table 2) that the

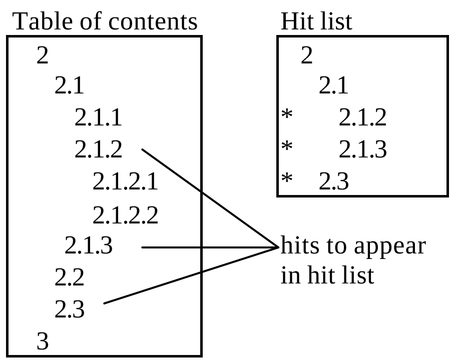


**Figure 3:** The hit list contains the headings of the text fragments that match the query (the hits) and the headings that lead in to the hits. Each heading consists of the section number and the heading text. The hits are marked with asterisks, whereas lead-in headings are unmarked.

| Level of lead-in viewer | LOGICAL | VENN | ALL |
|---|---|---|---|
| Level 0 (entire manual) | 0% (0 of 0) | 0% (0 of 0) | 100% (13 of 13) |
| Level 1 (chapter) | 0% (0 of 224) | 0.4% (1 of 226) | 33% (62 of 186) |
| Level 2 (subchapter) | 0% (0 of 581) | 1% (5 of 542) | 19% (86 of 447) |
| Level 3 (sub subchapter) | 0% (0 of 688) | 0.3% (2 of 609) | 3% (15 of 469) |
| Level 4 (sub sub subchapter) | 0% (0 of 85) | 0% (0 of 86) | 0% (0 of 45) |
| Total | 0% (0 of 1578) | 0.5% (8 of 1463) | 15% (176 of 1160) |

**Table 3**: Viewers of lead-in headings for the three retrieval modes that involve querying.

subjects used *View block* more than twice as often as *View fragment*, but was this to get more detail or more context? For the three retrieval modes that involve querying, Table 3 shows how often the subjects opened a lead-in heading to start reading at a level above the actual hits, presumably to extend them with more context. The hits from a query form a set of suggested entry points into the texts, and it is apparent that these suggested entry points have a strong influence on the subjects' information-seeking behaviour. In LOGICAL the subjects have not opened a single lead-in heading for reading; in VENN it has only happened a marginal number of times; and in ALL 15% of the viewers open a lead-in heading. The 184 lead-in viewers are opened one to four levels above the hits[1]. As much as 72% of the lead-in viewers are just one level above a hit, 22% are two levels above, 5% are three levels above, and 1% is four levels above. That is, the subjects use lead-in viewers sparingly or not at all, and when lead-in viewers are used it is typically to move just one level up relative to the hits.

Lead-in viewers occur across the majority of both tasks and subjects. Thus, lead-in viewers do not seem to be specific to particular tasks or particular subjects. It could be hypothesised that the subjects open lead-in viewers when they observe particular distributions of the hits. For each opened viewer we have calculated the number of hits that appear below it, i.e. in the text block starting with the viewed heading. The average number of hits below one of the 184 lead-in viewers is 2.35. This is slightly more than the average of 1.75 hits below the 4017 non lead-in viewers and weakly indicates that lead-in viewers are used when multiple hits cluster in different subparts of the same text block. However, further work is required to arrive at recommendations regarding how big a cluster of hits needs to be before retrieval algorithms should replace the individual hits with a hit that contains the individual hits by being at a higher level in the structure of the collection.

If lead-in viewers co-occur with extensive browsing it could further be hypothesised that the subjects utilise the browsing facilities to explore the hit list and thereby get interested in texts other than those returned as hits from the queries. This could explain why lead-in viewers appear in ALL but virtually not in LOGICAL and VENN, which include no browsing facilities. This explanation is, however, not supported by the data. For ALL, the proportion of browsing commands executed in the solutions containing a lead-in viewer is about the same as in the solutions without a lead-in viewer. We have not been able to find evidence for any of our tentative explanations of why lead-in viewers occur almost exclusively in ALL.

## 4.3 Entry Points versus Target Texts

Figure 4 shows how the hierarchical structure of the text collection can be used to divide the viewed texts into four groups depending on their position relative to the target texts. The target texts are the text fragments containing the answers to the tasks. Thus, the answer to a task is available to a subject whenever she/he views an on-target text and when he/she views an above-target text with *View block*. In the first case the target text has been identified by searching; in the second case the subject enters the text above the target text and attempts to identify it

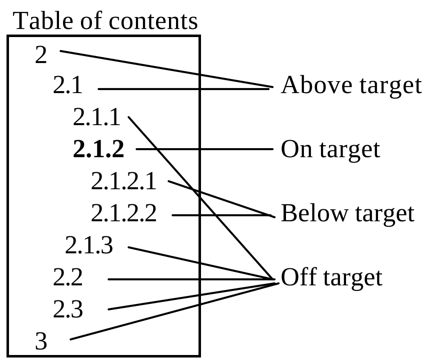


**Figure 4:** Definitions of *above target*, *on target*, *below target*, and *off target* relative to a target text, here text fragment 2.1.2.

[1] The depth of the document collection is five, so viewers cannot be more than four levels above a hit.

| | BROWSE | LOGICAL | VENN | ALL |
|---|---|---|---|---|
| At least one viewer is on target | 142 (46%) | 158 (51%) | 150 (49%) | 183 (58%) |
| Best viewer is above target, *View block* | 76 (25%) | 64 (21%) | 62 (20%) | 52 (17%) |
| Best viewer is above target, *View fragment* | 3 (1%) | 6 (2%) | 2 (1%) | 7 (2%) |
| All viewers are below target | 7 (2%) | 9 (3%) | 2 (1%) | 9 (3%) |
| All viewers are off target | 82 (26%) | 70 (23%) | 89 (29%) | 63 (20%) |
| Total | 310 (100%) | 307 (100%) | 305 (100%) | 314 (100%) |

**Table 4:** Solutions to the tasks, grouped according to the relationship between the opened viewers and the target texts.

by reading. Texts below or off target do not contain the answers to the tasks but may contain information that helps subjects subsequently find the target texts.

Table 4 shows the relationship between opened viewers and target texts. During 46-58% of the subjects' solutions of the tasks (row 1) the subjects viewed at least one on-target text. The subjects may have spent a long time finding this text and they may subsequently open additional viewers to consolidate their interpretation of it. These solutions are, however, evidence of an information-seeking behaviour where the target text was identified through searching. Contrary to this, 17-25% of the solutions (row 2) were reached by entering the texts above the target text and reading from there. During these solutions the subjects stopped searching before they had identified a target text and, instead, relied on their ability to quickly read a longer piece of text and spot the relevant parts. The remaining 25-31% of the solutions (rows 3-5) do not contain a single viewer that includes a target text. For these solutions the searching failed to identify the texts necessary for solving the tasks and the reading failed to enable the subjects to successfully redirect their searches. Consequently, the quality of these solutions is significantly lower than the quality of the solutions in the first two rows of Table 4: For the first two rows the average grade is 3.23; for the last three rows it is 1.75 (t-test, $p<0.0001$). There is no significant difference in task completion time (t-test, $p=0.23$).

The number of solutions that is based on opening an above-target text with *View block* is substantial for all retrieval modes. This suggests that users frequently need information about the context in which potentially relevant texts appear to be able to assess their applicability to the task at hand. Looking at the tasks for which the best viewer is above target, *View block* is much more frequent than *View fragment*. This may indicate that the subjects intentionally enter the text at a general level and from the outset expect to continue into the more detailed subparts of the text, once they have gained an overview. However, the subjects may also prefer *View Block* to defer the decision about whether to continue into the more detailed subparts of the text until after they have gained an overview. In both cases the subjects prefer to read without the interruption of – possibly – having to revert to searching to open another viewer.

Most tasks solved with LOGICAL, VENN, and ALL contain queries that return at least one hit that is below target (80-95% of the solutions depending on the retrieval mode). However, only 8-11% of the solutions contain a below-target viewer. Thus, in most cases the TeSS interface provides the subjects with enough information about the structure of the texts to avoid entering them at too detailed a level. A similar analysis for the *More* command, which expands a heading in the table of contents by adding the next more detailed level of subheadings, reveals that not a single *More* command is issued below target. As long as the subjects are navigating down through the structure of the text collection toward a target text it seems that the browsing facilities support them in effectively deciding when to stop searching and start reading. However, the large number of off-target viewers (all viewers are off target in 20-29% of the solutions, see Table 4) testify that information about the hierarchical structure of the text collection is not enough to direct the subjects down the right branches of the collection. This could indicate a need for facilities aimed specifically at supporting users in redirecting their searches.

# 5 Conclusion

Users of information retrieval systems intertwine searching and reading in order to find what they need and explore what is available. In this study, we investigate how 83 subjects intertwine searching and reading while using four different modes of a full-text retrieval system. We find that reading takes up at least one third of the subjects' task completion time, irrespective of whether they use a retrieval mode that offers browsing, querying, or a combination. During the middle part of the tasks – the period from the first to the last text viewer – the subjects spend more time reading than searching. Although browsing and querying are inherently different ways of seeking information, the use of one or the other retrieval mode generally seems to have only a limited impact on how the subjects incorporate reading into their information-seeking behaviour. Throughout the study our findings are very consistent across retrieval modes.

In accessing the text, the subjects show a clear preference for viewers presenting the whole subhierarchy of text belonging to a heading (i.e., they prefer the *View block* command). Such viewers are opened two to three times as often as viewers that contain the text belonging to a heading but exclude text belonging to its subheadings (i.e., the *View fragment* command). The subjects who use *View block* may from the outset intend to also read some of the more detailed subparts of the text, or they may be keeping this option open but deferring the actual decision until they have gained an overview of the text. By preferring viewers with text that spans multiple levels of detail, the subjects avoid an information-seeking behaviour where they alternate back and forth between short periods of searching and reading. This way the subjects achieve some continuity.

The subjects seldom enter the text via lead-in viewers, which can be utilised to turn from searching to reading at a level above the hits and thereby extend hits with more context. Thus, the *View block* commands are mostly used to get more detail, as opposed to more context. The subjects may perceive formulating a query and exploring the hits as one integrated process and, consequently, consider opening a lead-in heading as straying from the path they are currently pursuing. In effect, the hits returned from a query have a strong influence on the subjects' information-seeking behaviour. As a further indication of this, the majority of the lead-in viewers are just one level above a hit.

All four retrieval modes provide information that visualises the structure of the hierarchical text collection. This information aims primarily at providing the subjects with a feeling for the entire collection, and it seems to effectively support the subjects in avoiding to enter the text at a too detailed level. However, across the retrieval modes an average of 25% of the task solutions do not feature a single viewer containing the information necessary to solve the task. In these cases, the information about the structure clearly failed to enable the subjects to successfully focus and – subsequently – refocus their efforts.

A total of 51% of the tasks are solved by subjects who identify the answer to the task by searching and, thereby, reduce the scope of their reading. However, as much as 21% of the tasks are solved by subjects who enter the text at a higher level in the text hierarchy than the texts containing the answer and rely on reading from there. This strongly indicates that whereas reading is partly used as a subordinate to searching, it is also used in place of searching. Consequently, the texts containing the answers to the tasks may not be the best entry points into the text collection. Best entry points must strike a balance between searching and reading, and this study shows that the optimal balance is not achieved by assuming that the amount of reading should be minimised. Further studies are necessary to derive specific criteria for determining best entry points.

# Acknowledgements

This study was supported by the Danish National Research Foundation through its funding of the Centre for Human-Machine Interaction and by the British Royal Society through its funding of the GRIS project. The development of TeSS as well as the planning and execution of the experiment were done in collaboration with Jette Broløs, Marta Lárusdóttir, Kristian Pilgaard, and Flemming Sørensen. We wish to thank the students who participated in the experiment as subjects.

# References

Chiaramella, Y., Mulhem, P. & Fourel, F. (1996), *A model for multimedia information retrieval*, Technical Report Fermi 4/96, ESPRIT BRA Project N. 8134 (Available as http://www.dcs.gla.ac.uk/fermi/tech_reports/reports/fermi96-4.ps.gz).

Cleveland, D.B., Cleveland, A.D. & Wise, O.B. (1984), Less than full-text indexing using a non-Boolean searching model, *Journal of the American Society for Information Science* 35(1), 19-28.

Dix, A.J., Finlay, J.E., Abowd, G.D. & Beale, R. (1998), *Human-Computer Interaction*, Prentice Hall.

Frøkjær, E., Hertzum, M. & Hornbæk, K. (2000), Measuring usability: Are effectiveness, efficiency, and satisfaction really correlated?, in *Proceedings of the ACM CHI 2000 Conference on Human Factors in Computing Systems*, ACM Press, pp. 345-352.

Furnas, G.W. (1999), The fisheye view: A new look at structured files, in S.K. Card, J.D. Mackinlay & B. Shneiderman (eds.), *Readings in Information Visualization: Using Vision to Think*, Morgan Kaufmann, pp. 312-330. (Reprinted from *The fisheye view: A new look at structured files*, Bell Laboratories Technical Memorandum #81-11221-9, October 12, 1981).

Hearst, M.A. (1995), Tilebars: Visualization of term distribution information in full text information access, in *Proceedings of the ACM CHI'95*

*Conference on Human Factors in Computing Systems*, ACM Press, pp. 59-66.

Hertzum, M. & Frøkjær, E. (1996), Browsing and querying in online documentation: A study of user interfaces and the interaction process, ACM Transactions on Computer-Human Interaction 3(2), 136-161.

Lalmas, M. & Moutogianni, E. (2000), A Dempster-Shafer indexing for the focussed retrieval of a hierarchically structured document space: Implementation and experiments on a web museum collection, in *Proceedings of the RIAO 2000 Conference on Content-Based Multimedia Information Access, Vol. I*, Center of Advanced Study of Information Systems, pp. 442-456.

Robertson, G.G., Card, S.K. & Mackinlay, J.D. (1993), Information visualization using 3D interactive animation, *Communications of the ACM* 36(4), 57-71.

Tenopir, C. & Ro, J.S. (1990), *Full Text Databases*, Greenwood Press.