

Interactive File Searching Using File Metadata and Visualization

Pedro Teixeira Daniel Gonçalves Manuel J. Fonseca
Dept. of Computer Science and Engineering
INESC-ID/IST/TU Lisbon, Lisbon, Portugal
pedro.lopes.teixeira@gmail.com, {daniel.goncalves,mjf}@inesc-id.pt

ABSTRACT

Navigation and browsing on a computer system are usually done using the file system hierarchy. However, this is not the most adequate method to search or locate a given file at a later time, unless we know exactly where it is. In this paper, we present a new approach for interactive file searching, which takes advantage of the (implicit) metadata associated to files. With this information we specify a semantic hierarchy that is both used for browsing and filtering the found files. We combine this with an interactive histogram, for overiewing and filtering according to specific metadata, and a dynamic view mechanism, which selects the most appropriate representation for the type of file being presented. Experimental tests with users revealed that they can find files faster than with a conventional browser or searching tool and that the semantic hierarchy and filtering mechanism were well understood by users.

Categories and Subject Descriptors

H.5.2 [Information Interfaces and Presentation]: User Interfaces—*Graphical User Interfaces*

General Terms

Design, Human Factors

Keywords

File searching, file metadata, interactive visualization, semantic hierarchy

1. INTRODUCTION

It is nowadays common for users to handle thousands of electronic documents, which are stored in a hierarchically organized file system. When trying to retrieve a file using a traditional filemanager, users can resort to little more information than the file's location in the hierarchy. However, such a classification is fraught with problems. When storing a document, where to place it is often not a trivial decision. More than one place in the hierarchy (or no place at all)

might seem adequate. Also, what seems a good classification at one point in time might not be the one remembered at a later time.

Users can also find their files resorting to desktop searching tools (e.g. Google Desktop, Apple Spotlight or Windows Desktop Search), where they can use the file name, some text contained on it or its attributes. These type of tools are adequate and provide good results when we know the name of the file or some words belonging to it. However, when we want to find a file from which we do not know the name or/and that does not contain text (a picture or a music for instance) its location is not immediate, requiring users to compose a set of complex filters to achieve meaningful results. Another drawback of these solutions is the fact that they only focus on searching, discarding the presentation of results. Typically, results are handled by the file manager of the operating system, or are presented as a list of file names. This list is suitable when we have few results, but it makes the location difficult if the list is very long.

In this paper, we present a novel approach that allows users to easily and efficiently locate files and browse the results, forgetting about the folders where they are stored. To that end, we combined three main features. First, we defined a semantic hierarchy created using the metadata associated to files (e.g. file type, modification date, size, author, number of pages, etc.) that allows users to interactively filter results by combining various values. Second, we created an interactive histogram, which provides users with an overview of their files and additionally allows them to filter the listed files interactively through selections on it. Finally, we developed dedicated views for each type of file, thus providing users with the most useful information at each time. The traditional searching mechanism based on keywords was also integrated with these three characteristics of our solution.

To develop our approach, we started by first performing a user study to understand how users browse and search files using current applications, and also to identify the knowledge that users have about their files and their file system. Based on this, we identified a set of design implications, which we materialized on a prototype for interactive searching of files regardless of their location on the file system. Experimental evaluation with users showed that they understood well the semantic hierarchy and the filtering scheme, allowing them to locate files faster than with the Mac OS X Finder searching mechanism.

The contributions of this research are a new approach for interactive file searching using metadata and visualization, and the comparative evaluation of the resulting prototype. The searching approach is based on a semantic hierarchy created from the metadata associated to files and on an interactive histogram to provide an overview of the user's file system. Both quantitative and qualitative results of the study confirmed the utility of our approach.

In the remainder of the paper, we present and analyze in Section 2 some related works on file browsing, trying to understand what failed on trying to free users from the directory exploration. Section 3 presents the main results of the study with users and the design implications identified. Next, we describe our solution based on a semantic hierarchy for interactive searching of files, detailing each of the three main components. In Section 5 we present results from the experimental evaluation with users. Finally in Section 6 we conclude the paper and discuss future work.

2. RELATED WORK

To help users manage their files and documents several paradigms for file browsing and exploration were developed allowing the visualization of users' document collections in meaningful ways. These approaches, by moving away from the file system hierarchies, strive to convey an overall view of the users' documents. In this section we describe some of the solutions developed to tackle this problem.

PhotoMesa [2] is a solution for image browsing that relies on metadata (e.g. date, local, people, etc.) to group photos. It uses Treemaps [7] as the main visualization technique, flattening down folders and sub-folders to the same level. This lack of leveling increases the number of files shown at the same time, which is good for a small number of files but becomes unusable for larger collections (e.g. a file system). Although, PhotoMesa is a good browser for Photos, it is quite difficult to apply this kind of visualization and browsing to all file types.

PHLAT [4] is an interface for Windows Desktop Search that combines browsing and searching in one unique interface, through the creation of textual queries and filters. The user interface of PHLAT has the typical look from "Microsoft Office" applications, providing only text-boxes and checkboxes for users to filter, but not offering any special visualization technique to give an overview of the returned results. This kind of interaction does not distinguish much PHLAT from common search applications, such as Live Search, Spotlight, Google Desktop, etc., which rely mainly on the name of the file and on its content.

FacetMap [8], a faceted browser designed for searching and browsing tasks combines metadata from files to define the exploration tree, and uses Treemaps as the main paradigm for visualization and filtering, allowing users to browse and search files in an easy way. Although, this idea of combining metadata and filtering is very interesting, the lack of usability and the reduced number of files that it can show at the same time are drawbacks of this approach.

Leap [9] is a Mac OS X application, which uses Spotlight as basis to browse and search files. Leap uses mainly file tags

(added manually) and file metadata to filter and organize results. However, contrary to our solution, there is no relationship and organization between tags/metadata, causing users to easily lose context and get lost in a collection of non-related metadata.

In addition to these solutions, there are others that try to create new ways for browsing and searching files, but they continue to be based on the directory hierarchy. Discovery [1] is a Treemap-based application, which shows the entire file system in the screen at the same time, using colors to distinguish different file types. StepTree [3] is basically similar but shows the information in 3D. Liquifile [10] is a file manager for Mac OS that besides showing the directory/file tree, draws circles to convey information about file size, and uses their position to inform about the creation date.

In the context of document location we have Stuff-I've-Seen [5] that indexes all information elements handled by the user, allowing them to be retrieved using a keyword-based search mechanism, and filtered using the available metadata. MyLifeBits [6] is another approach that aims at being able to automatically record all information relevant for any given user. It uses a database to store content of different types, such as contacts, documents, email messages, events, photos, music and video, each with its own metadata properties. Although, these are interesting solutions, they are focusing more on providing users with information about their documents and the relationship between them to help users navigate their "bits" of information in search of a specific one.

Looking at the majority of the existing solutions, we can see that none of them succeeds in providing the user with a solution to browse and find files regardless of their location in the directory tree. Although, there are some approaches using metadata from files and tags, they do not take advantage of a metadata-oriented organization and visualization; do not offer efficient and easy to use filtering mechanisms; and do not provide distinctive views for different file types.

Our approach tries to combine all of these, to produce an efficient solution for interactive file searching, based on the semantic information associated to files, and by providing a filtering and overview mechanism based on histograms.

3. USER STUDY

To characterize the potential users of our solution and to understand how they currently execute browsing and searching tasks, we carried out a user study by performing an online survey and a contextual inquiry. To support these we created a questionnaire to be answered by potential users. The online survey was disseminated through the university mailing list and the in person inquiry was performed to a set of selected people to be representative of the potential users.

We received 78 answers to the online survey and performed eight contextual inquiries, mainly to validate the results obtained from the survey. During the personal inquiries we asked users to answer the same questionnaire from the online survey and to execute some typical tasks of browsing and searching, while we collected information. At the end, we did an informal interview to clarify some points identified during task execution. Here we present a summary

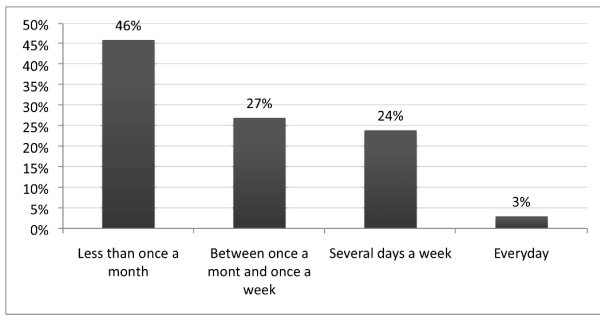


Figure 1: How often users do not know the location of a desired file.

of the main outcomes from the online survey and from the contextual inquiry.

The majority of the users were between 18 and 30 years old (67%) being 32% females and 68% males. Almost all users (88%) use the computer every day and had some experience with computers, like browsing files, read the email and surfing the web, but they had different backgrounds and skills. We verify that almost all users (99%) use the OS default applications for browsing (e.g. Mac OS Finder, Windows Explorer, etc.) and for searching (e.g. Spotlight, Windows Desktop Search, etc.).

We tried to find out how often users do not know where a desired file was located. As we can see from Figure 1, almost half of the users (46%) stated that they do not know where a file is less than once a month, 27% between once a month and once a week, 24% several days a week and 3% every day. Although, 46% of the users are optimistic and know almost always where their files are, 27% do not know the location of a file several days a week.

From the study we noticed that the file name is the characteristic that users considered most important to use and to remember. We agree that the file name is very relevant for the browsing and searching of files, however, users should use it through recognition and not through recall, as the current searching solutions oblige.

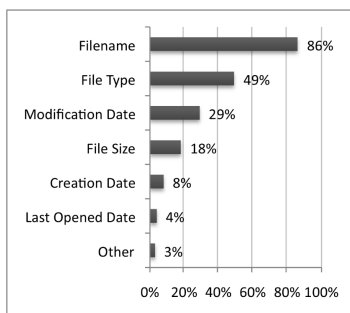


Figure 2: Characteristics used to sort files while browsing.

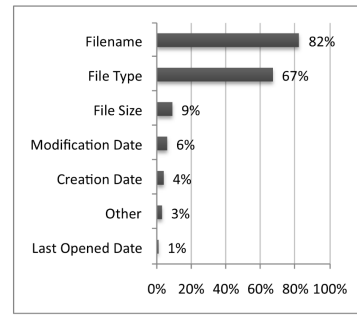


Figure 3: Characteristics remembered while searching for a file that users do not know the location.

With regard to file characteristics, we observed that the File Type came in the first place as the most used characteristic for sorting files in a folder (49%) and as the most remembered metadata when users want to search for a file that they do not know its location (67%). The File Type is followed by the Modification Date (29%) and File Size (18%) while sorting and by File Size (9%) and Modification Date (6%) while searching, as depicted in Figures 2 and 3. On average we can put File Type on first (58%), followed by the Modification Date (18%) and the File Size (14%).

Finally, and since File Type was considered the more relevant metadata by users, we tried to check if they were aware of the most common file types. We noticed that almost all users (around 90%) know Text, Picture, PDF, Email, Audio, Presentation and Video files. During the in person questionnaires, we asked users for examples of file types they knew. Almost all of them gave examples of applications and file extensions that were correctly related to the given file types.

From the data collected during the online survey, and validated by the contextual inquiry, we were able to identify some relevant outcomes that influenced the design of our solution. First, we noticed that users have some experience using computers and know the name of almost all common file types. This is relevant, since we observed that users consider the File Type as the most important metadata, followed by Modification Date and File Size, when browsing and searching for files. Second, we verify that a quarter of the users (27%) found themselves looking for a file that they do not know where it is, several days a week. In the next section, we present our solution that took these design implications into consideration.

4. PROPOSED SOLUTION

With our solution we wanted to answer the following question: “How can users efficiently and easily locate a file without knowing where it was stored?”. To that end, we first tried to find if this question was legitimate by performing a user study. From the results we design a solution for interactive file searching that replaces the directory structure by a semantic hierarchy, created using the files metadata. This hierarchy is used not for organizing files, but to organize the different metadata extracted.

Moreover, we take advantage of the knowledge that users

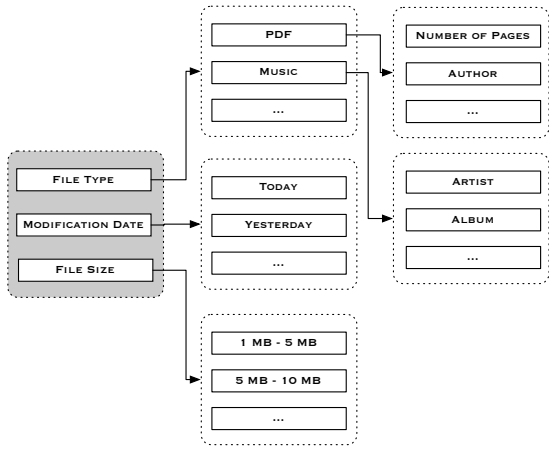


Figure 4: Structure of the Semantic Hierarchy.

have about File Types, making it the main filter of our searching mechanism. Additionally, we combine this with an interactive histogram to give an overview of the existing files and to allow users to perform fast filtering actions. Finally, and since our solution performs a first filter by file type, we provide different views for different file types, making the presentation of files richer and consequently easier for users to recognize rather than recall what they are looking for.

4.1 Semantic Hierarchy

The main objectives of this hierarchy, unlike the hierarchy of directories, are to present, organize and filter the metadata from files, creating a semantic organization instead of a spatial organization. This hierarchy is based on the fact that different file types have different metadata. Thus, for each file type we have different properties that can be combined.

4.1.1 Creation and Metadata

Each file type has certain features that characterize it. For example, a picture can be oriented in landscape or portrait, a music can have a duration of more than five minutes or a PDF can have a certain number of pages. However, it does not make sense to say that a PDF has a duration or a song is in landscape.

To identify what metadata was associated with each file type, and since we are using Spotlight¹ as the searching mechanism, we examined the information that Spotlight is able to extract from each file type. Based on this and on the results from the users study (see Section 3) we created the semantic hierarchy with three nodes on the top level: File Type; Modification Date and File Size (see Figure 4 left).

Modification Date will have sons such as “Today” and “Yesterday”, and File Size will have several intervals for size (e.g. “1 MB - 5 MB”, “5 MB - 10 MB”, etc.).

The File Type node has two more levels, one with the differ-

¹Spotlight is a searching mechanism, similar to Google Desktop, offered by Mac OS X.

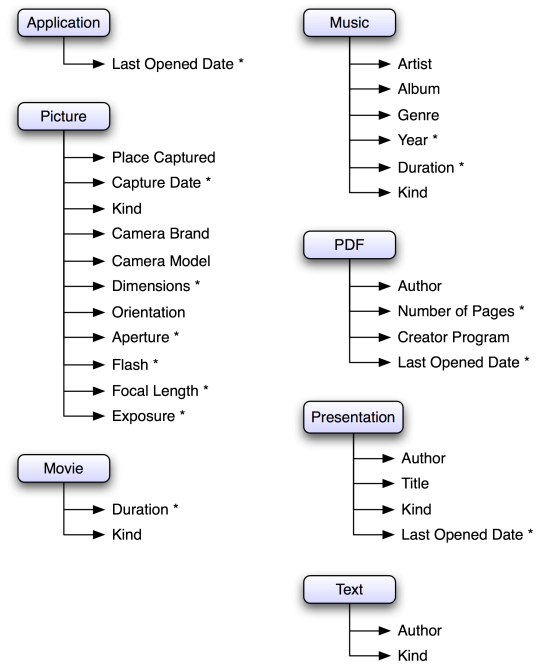


Figure 5: Semantic Hierarchy for the different file types. Metadata marked with an (*) can be used in the interactive histogram.

ent file types and another with the specific metadata associated with each file type. This set of file types and their metadata were selected according to the information collected during our user study (see Figure 5). Currently we have nine file types in this set plus an “Other” field where users can write the name of any file type (see Figure 6 left). In the future we can include new file types and new metadata, since our solution is flexible to accommodate that change.

As we can see, our semantic hierarchy has a maximum of three levels, making the browsing and searching actions faster since it reduces the number of potential clicks. Moreover,

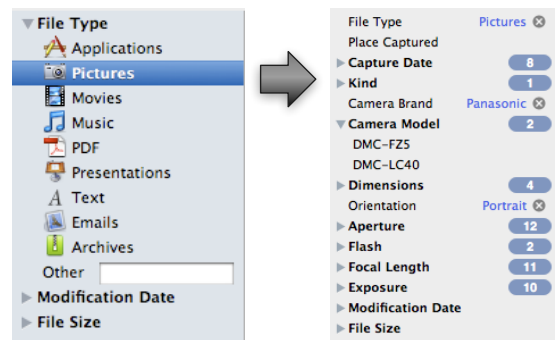


Figure 6: Semantic Hierarchy after selecting Pictures as File Type.

and contrary to other applications (e.g. Leap), our solution creates this fixed hierarchy of metadata that always presents relevant and related information to the users, offering an organized way of browsing their files. The users never get lost and always have a context of the current state, being able to change it whenever they want by interacting with the semantic hierarchy.

4.1.2 Interactive Searching

One of the major limitations of existing applications such as Spotlight, Windows Search, Google Desktop, etc., is that they focus only on the search discarding the information that they have about the files stored in the file system. As a consequence, users have to create their queries from scratch without any help from the searching tool. For example, these tools let users search for photos taken with a "Casio" camera, "Heavy Metal" music or AVI movies, when they already know that there are no files that satisfy this.

Our solution, besides offering the same searching mechanism by keywords, also helps users while they are searching for a specific file, informing them before hand about the existing files. It provides constant and actual information about users' files relevant to the current search. This way, users can interactively explore the metadata and create meaningful search queries. Users can for instance browse their music or photos through their properties: know what genres or from what years their music are, what cameras were used to take pictures, what files are occupying more space in the disc, etc.

The interactive search is performed by selecting values or ranges of values for the files metadata presented in the semantic hierarchy. This hierarchy is drawn like a tree directory, where folders correspond to the metadata and files within each folder correspond to the different values for each metadata (see Figure 6 right). Users can begin their search/exploration by choosing a file type, a range of modified date or a range of file size. Any of these can be combined while searching for files.

If the user selects Modification Date or File Size, the system shows the range values for each metadata, while for File Type it lists the various types of file. After selecting a file type the application returns the files of that type and presents the number of different values for each file property (e.g. Kind, Camera Brand, etc.), as illustrated in Figure 6 right. These values are automatically updated whenever the user chooses values for the various files properties. For example, in Figure 6 right we can see that "Portrait" "Pictures" were taken with two "Panasonic" Camera Models, in eight different Capture Dates and using twelve Apertures. So, if we now select one of the Camera Models (DMC-FZ5 or DMC-LC40), the list of files will be reduced to those that satisfy the current values of the metadata and the number of values for the other metadata are automatically updated.

In summary, our semantic hierarchy, not only organizes the metadata, but also gives information to users about the number and of the different values that exist for each metadata. This way users get information about their files and can perform interactive and iterative searches with knowledge about their file system.

4.2 Interactive Histogram

Our approach includes also a new method with a dual functionality, the interactive histogram. First, it provides an overview of the existing files showing how they are distributed along the metadata. Second, it offers an interactive mechanism to specify intervals of values that are used to filter the results. This specification is more flexible and powerful than the ordinary text-based mode and provides instant results. The histogram represents continuous metadata, such as Modification Date, File Size, Number of Pages, Music Duration, in the form of a chart, where the x axis is the metadata value and the y axis represents the number of files, as illustrated in Figure 8 area 2.

4.3 Dynamic Views

The majority of the existent applications present the found files as an ordinary list and in some cases using thumbnails for images.

Considering that one image is easier to recognize in the form of a thumbnail, music through its name, artist, album or genre, and a PDF document by the number of pages, we decided to develop different views for different file types. Our decision was supported by the results collected from the user study, where we found out that users considered the File Type as the main metadata. Moreover, and since our overall solution is File Type oriented, we assume that users start browsing and searching by first selecting the File Type. This way, we can choose the most appropriate visualization for the present situation. In the case that users do not select the File Type, we present a generic list view.

4.4 Architecture and Implementation

We developed our final prototype of the Magoo application for the Mac OS X, using the Cocoa development environment and the Spotlight library. Our system is composed by two main modules, the Filter and the Dynamic View engines (see Figure 7). The former is responsible for the presentation of the semantic hierarchy, for the interactive selection of metadata values and for the histogram. The latter deals with the different views and applies the selection performed on the histogram.

While searching for files the user selects values for various metadata on the semantic hierarchy and/or type some keywords on the search field. These selections are converted into a query to the Spotlight search engine. Results returned by

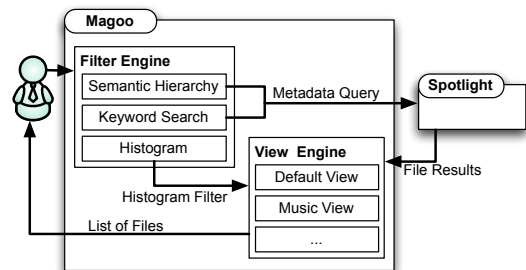


Figure 7: System architecture.

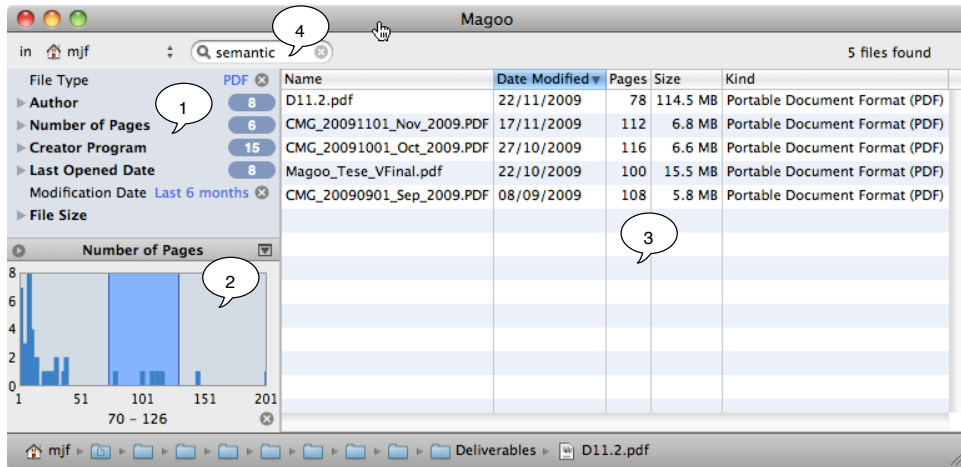


Figure 8: Prototype for browsing and searching files using semantic hierarchy and the interactive histogram. In this example the user is searching for a PDF file containing the word “semantic”, modified during the last six months and with 70–126 pages.

the Spotlight are processed by the Dynamic View engine, filtered according to the values specified in the histogram and presented using the right view for the current file type.

The resulting application has three main areas, as illustrated in Figure 8. Area 1 presents the semantic hierarchy, where users can first select by File Type, Modification Date or File Size. After selecting one of these options the system shows the next level of the hierarchy. In the current example, the user selected “PDF” as File Type, and the system showed the main metadata associated to PDF (Author, Number of Pages, Creator Program, Last Opened Date) indicating the number of occurrences for each characteristic. The user also selected the Modification Date (Last 6 months). Area 2 contains the histogram, showing the number of PDFs per Number of Pages. With this type of visualization, users can get an idea of the distribution of PDFs by their sizes in pages and they can also select one region to filter the results. In the example we can see that the user has more PDF files (modified in the last 6 months) with a small number of pages, but also has a file with 201 pages. Finally, area 3 lists all the files that satisfy the different selections and filters applied by the user. Users can also combine the traditional textual search (area 4 in Figure 8) with the new mechanisms such as the semantic hierarchy and the interactive histogram.

5. USER EVALUATION

The main goal of our solution is to provide users with an efficient and easy mechanism to find files without knowing their location. To check whether our objectives have been met, we conducted an experimental evaluation with users to evaluate the usability of our tool. We compared it against the searching mechanism included in the Mac OS X Finder, using both objective and subjective measures. The objective measure was the time to complete a set of tasks, while the subjective measure was the users’ satisfaction with the tools. We measured it through a questionnaire with a set of tool-specific questions, like for instance, the difficulty to locate a

file, the easiness to perform the tasks, the understanding of the semantic hierarchy, the use of the histogram or the use of the File Type as the main filtering element.

We selected the Mac OS X Finder because it is the searching (and browsing) application that comes by default with the operating system, and as we found out in the users study, 99% of the users use the default tool. The Finder/Spotlight application allows users not only to browse files as a common file manager, but it also permits the creation of searching queries using the combination of different metadata (see Figure 9). The metadata supported by the Finder is the same as Magoo, since both use the Spotlight search engine.

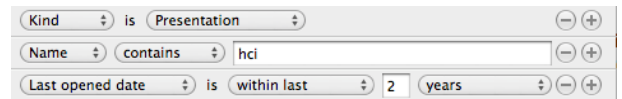


Figure 9: Composition of a query using the Mac OS X Finder/Spotlight.

5.1 Participants

Ten volunteer users (nine male and one female) aged between 18 and 30 years, belonging to the set of potential users, participated in our experimental evaluation. Nine of them had attended the university (or were attending at the time of this evaluation) in fields as diverse as architecture, medicine, economics, education, literature and engineering, with only one person in an area related to computing. The other user was still in high school.

All users used the computer every day, except one that used it a few times a week. Moreover, nine users used the Windows XP or Vista operating system, while only one user uses the Mac OS X. We decided to choose mostly users who were not familiar with the Mac OS X, so this way all of them would have the same level of experience with the two

applications and we can assess their learnability.

5.2 Procedure

We performed individual tests with each user, using a within-subject design, on the same computer and using the same set of files (more than 300,000 files). This way all users were in the same situation, all of them search for the same files and we have a ground truth to compare the achieved results.

The experimental evaluation consisted of three phases. First, the observer described the objectives of the evaluation and explained the functionalities of the two applications, answering any question raised by the participant.

Next, we delivered the list of five tasks on paper (as depicted in Table 1) to the participant who performed each of these tasks in the two applications. The participant performed the five tasks on one application, and only then he moved to the other system. Before starting any task, the applications were restarted, so all users started their tasks with the tools in the same state. To counterbalancing conditions, we changed the order of the applications across users. Finally, users answered the satisfaction questionnaire.

5.3 Tasks

We defined five tasks to be performed by all users on each of the two applications. The tasks were designed to represent situations from our everyday life. Some of them had already happened to us, others were identified during the contextual inquiry, when we talked to users. We varied the kind of tasks, by using different types of files (text, music, pictures and video) and different mechanisms to locate the files, browsing (on Task 1) and searching (on Tasks 2–5).

Table 1: Tasks used during the user evaluation.

Tasks	Description
Task 1	Visualize all the music in the computer from the 90’s, and annotate the year with more music.
Task 2	Locate a PDF paper with 10 to 20 pages that talks about “fisheye view” and that was stored in the computer between February and June 2008.
Task 3	Find pictures of a dog, taken at the beach in January, with a Panasonic camera.
Task 4	Find a document written with MS Word that contains “afraid of the dark”.
Task 5	Search for the last Woody Allen movie (Vicky Cristina Barcelona) that was stored in the computer two weeks ago.

5.4 Results

Each individual session took up to one hour. In general, almost all participants performed faster in our application. Moreover, they understood well the functioning mechanism behind our approach and considered the execution of tasks easier on our tool.

5.4.1 Execution Time

The average execution time for the five tasks is smaller in our application than in the other system as depicted in Figure 10. A t-test with pairwise samples revealed that the average

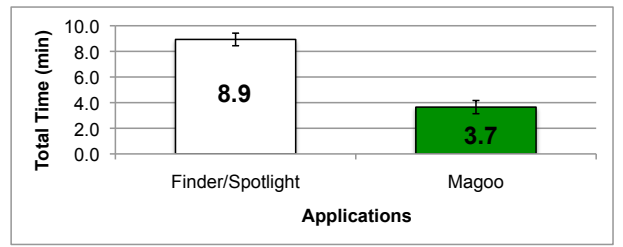


Figure 10: Total execution time for each application, with standard deviation bars.

time taken using Magoo ($mean = 3.7min$) was significantly smaller than using Finder/Spotlight ($mean = 8.9min, p < 0.01$). This means that on average users took 141% more time on Finder/Spotlight to complete the five tasks than on our prototype, showing that our solution achieved the goal of being more efficient.

We also analyzed the time that users took to perform each task on the two applications. Our goal was to understand how each system behaves for the different kind of tasks (browsing or searching) and for the different types of files (text, music, video, images).

As we can see from Figure 11, on average, users always performed faster on our system than on the Finder/Spotlight application. The pairwise t-test shows that the average time taken for each task is significantly smaller on Magoo than on Finder/Spotlight, with a value of $p < 0.03$ for all tasks.

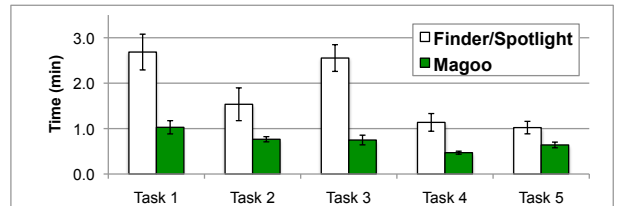


Figure 11: Execution time per task for each application, with standard deviation bars.

We can also notice that tasks which require more complex queries to achieve the results, such as tasks 1 and 3, users took 60% less time in Magoo than on Finder/Spotlight. These values show that our goal of providing a tool for interactive browsing of files regardless of their location on the folders was achieved.

5.4.2 User Satisfaction

Our satisfaction questionnaire was divided into three parts. One served to characterize users about their experience with computers and about the operating system that they use. The second part was used to identify the easiness of use of the two applications. Finally, the last part tried to validate some of the decisions that we took during the development of our solution, such as the use of the semantic hierarchy, the use of the File Type as the main filtering metadata and the use of the interactive histogram to provide overview and filtering functionalities.

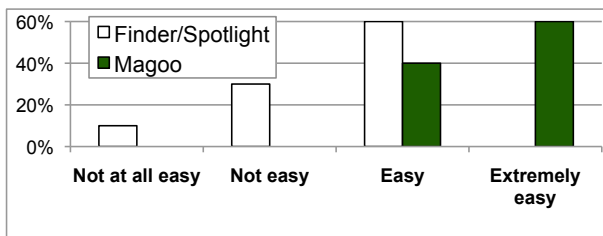


Figure 12: Finding the desired files was ...

From the satisfaction questionnaire we found that users considered searching for files and the execution of the five tasks easier on our application than on Finder/Spotlight, as illustrated in Figure 12. Indeed, all users considered that the execution of the five tasks was easier on our tool, and when we asked to select one of the tools as the easiest one to use, all of them selected ours.

In the next part of the questionnaire, participants rated the main elements included in our solution, namely the semantic hierarchy, the histogram and the use of the file type as the main metadata. Results show that 20% of the users considered that it was easy and 80% extremely easy to understand the interactive searching and filtering mechanism based on the semantic hierarchy. For the use of the interactive histogram, 70% considered it easy and 30% extremely easy. Some users, suggested to increase the size of the histogram, because sometimes it was difficult to specify exact values. Finally, 90% of the users strongly agree with the use of the File Type as the first step in the searching/filtering process.

During the experimental evaluation we also collected some comments from users:

“Even if I know where the file is, it [Magoo] takes me less clicks to find it than to navigate to the right place in the hierarchy.”

“With this system [Magoo] I do not need to remember the names of all my files.”

“Although, sometimes it is hard to specify an exact value, the histogram is very intuitive to use and gives me a good overview of my files.”

From these results, we can conclude that users clearly understood the semantic hierarchy, the use of the histogram and agree with the choice of the File Type as the first option to use during filtering operations.

6. CONCLUSIONS AND FUTURE WORK

In this paper we presented a new approach for interactive file searching based on a semantic hierarchy created from the metadata extracted from files. Our solution hides the traditional directory structure, allowing users to forget about the location of their files and concentrating only on pertinent information about them.

Our solution takes advantage of the specific metadata associate to each type of file for presenting the next levels of the semantic hierarchy and to decide which visualization mechanism is used to show the list of files resulting from the interactive search. This is a major advantage of our ap-

proach, since existing solutions do not explore this relationship between metadata, leading users to situations where they easily get lost and out of context.

To achieve this solution we first performed a user study to understand how users perform tasks related to browsing and searching for files, and what is the knowledge that users have about their files and their file system.

Experimental evaluation with users revealed that users were able to browse and find files faster using our solution than with the other application. Additionally, users understood very well the semantic hierarchy and the filtering mechanism based on the interactive histogram that we introduced in our approach.

We are currently preparing a long-term experiment of two to three months to be performed in the users’ computers using their files, to see if users change their way of storing and naming files, due to the use of our application.

Acknowledgments

This work was supported by FCT (INESC-ID multiannual funding) through the PIDDAC Program funds and through the Crush project , PTDC/EIA-EIA/108077/2008.

7. REFERENCES

- [1] T. Baudel. Browsing through an information visualization design space. In *Conf. on Human Factors in Computing Systems (CHI’04)*, 2004.
- [2] B. B. Bederson. Photomesa: a zoomable image browser using quantum treemaps and bubblemaps. In *ACM Symposium on User interface Software and Technology (UIST’01)*, 2001.
- [3] T. Bladh, D. A. Carr, and J. Scholl. Extending tree-maps to three dimensions: A comparative study. In *Asia Pacific Conference on Computer-Human Interaction (APCHI’04)*, 2004.
- [4] E. Cutrell, D. Robbins, S. Dumais, and R. Sarin. Fast, flexible filtering with phlat. In *Conf. on Human Factors in Computing Systems (CHI’06)*, 2006.
- [5] S. Dumais, E. Cutrell, J. Cadiz, G. Jancke, R. Sarin, and D. C. Robbins. Stuff i’ve seen: a system for personal information retrieval and re-use. In *Conf. on Research and development in information retrieval (SIGIR’03)*. ACM Press, 2003.
- [6] J. Gemmell, G. Bell, and R. Lueder. Mylifebits: a personal database for everything. *Communications of the ACM*, 49(1), 2006.
- [7] B. Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics*, 11(1), 1992.
- [8] G. Smith, M. Czerwinski, B. Meyers, D. Robbins, G. Robertson, and D. S. Tan. Facetmap: A scalable search and browse visualization. *IEEE Transactions on Visualization and Computer Graphics*, 12(5), 2006.
- [9] I. Software. Inc., leap 1.1. <http://www.ironicsoftware.com/leap/index.html>, 2009. as seen in February 2009.
- [10] C. Waldeck. Liquid 2d scatter space for file system browsing. In *Conf. on Information Visualisation*, 2005.