ABSTRACT

Title of Dissertation: MANAGING AND EXPLORING PERSONAL
MEDIA USING SEMANTIC REGIONS: A
SPATIAL INTERFACE SUPPORTING USER-
DEFINED MENTAL MODELS

Hyunmo Kang, Doctor of Philosophy, 2003

Dissertation directed by: Professor Ben Shneiderman
Department of Computer Science

Computer users deal with large numbers of personal media objects such as images, audio clips, voice mails, video clips, web pages, emails, and various document files. Users often struggle to interpret, explore, arrange, and use personal media objects because of three major problems; an ever increasing amount of personal media data, rigid organizing metaphor, and difficulty in rapid data access.

With the progress of computer hardware and digitization technologies at the rate of Moore's law, users may generate, acquire, and store more and more personal media data on their personal machines. However, the means available for users to organize and customize their personal media information spaces are extremely poor and driven mostly by storage and distribution models, not users' needs. Furthermore, this rigid and system-oriented metaphor causes wide and deep file folder hierarchies and often forbids users to rapid data access from storage.

This dissertation uses a graphical mechanism for spatially organizing personal media data based on users' mental models and introduces an innovative interaction

metaphor to apply users' mental models to personal media data. *Semantic Regions* introduces an innovative way to construct users' mental models by drawing regions on 2D space and specifying the semantics for each region so that users can apply various personal ontologies to personal media data using the *fling-and-flock* metaphor. The prototype application, MediaFinder, validates the usability of the interface, particularly in comparison with alternative approaches. Contributions of this dissertation include:

- *Semantic Regions* provides a formal model of spatial and dynamic reorganization of personal media data based on users' mental models. It extends the concept from a system-oriented file management system to a user-oriented personal media management system by employing the semantics of personal media data.
- The MediaFinder personal media management tool, which uses *Semantic Regions*, the *fling-and-flock* metaphor, and flexible categorization capabilities to explore and manage personal media data sets. MediaFinder's object-oriented architecture can easily be extended to support variants of the *Semantic Regions* model.
- The design and implementation of interaction models that support the personal media management tasks such as organization, meaning extraction, navigation, search, indexing, and navigation.
- Two usability studies provided preliminary insight into the utility of *Semantic Regions* and led to design improvements for the construction and operation of *Semantic Regions*.
- A framework for extending the *Semantic Regions* model, including the descriptions of possible extensions.

MANAGING AND EXPLORING MEDIA USING SEMANTIC REGIONS: A SPATIAL INTERFACE SUPPORTING USER-DEFINED MENTAL MODELS

By

Hyunmo Kang

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2003

Advisory Committee:

Professor Ben Shneiderman, Chair and Advisor

Associate Professor Ben Bederson

Associate Professor Kent Norman

Associate Professor Adam Porter

Associate Professor Amitabh Varshney

DEDICATION

To

My Parents

And

Keeyoung

ACKNOWLEDGEMENTS

Most of all, I have to thank Ben Shneiderman for his advice, encouragement and inspiration. Without his invaluable support and patient encouragement, this dissertation would not exist. His encouraging words inspired me to continue whenever I was in trouble with the direction of my research. Working with him has been a truly amazing experience, I have been always amazed at his enthusiasm for research as well as his wide knowledge about human and computer. I could not possibly list all that I have learned from him and been influenced by him.

Thanks to my dissertation committee members, Ben Bederson, Kent Norman, Adam Porter, and Amitabh Varshney for many suggestions that have improved this dissertation. Ben Bederson showed me a new world in Computer Science. In my course work years, I was fortunate to take his class and thus I could get a chance to become aware of what HCI was and why this field would be important for me and my country. He also provided interesting discussions and invaluable advice regarding GUI design and implementation as well as different perspective on Information Visualization.

I also thank to all the members of Human Computer Interaction Lab members whom I have been pleased to work with. All the faculty members; Ben Shneiderman, Ben Bederson, Catherine Plaisant, and Allison Druin have made the HCIL a great place to work. My special thanks go to Anne Rose. I will not be able to forget her smiling face and her efforts to protect me from the meticulous CS staffs. Chris North, Egemen Tanin, Harry Hochheiser, Juan Pablo Hourcade, Lance Good, Jaime Montemayor, Bill Kulles, Bongwon Suh, Jinwook Seo, Bongshin Lee, Hilary Browne Hutchinson, and other HCIL students have provided a supportive and engaging working environment.

I greatly thank those who funded my research, including Intel, Microsoft, IBM, ACM SIGCHI, Ricoh, and Census Bureau. Thanks to Greg Wolff and Peter Hart at Ricoh Innovations Inc. for their support and interesting discussions on my research topics.

Thanks also to my precious Korean colleagues in Computer Science department. They have been always my supporters and we have shared our joy and pain as foreign students studying in US.

My six years of Maryland life has been enriched by the special friends I met here in University of Maryland at College Park ; Woongtae Kim in Astronomy, Hyuncheol Koo in Electrical and Computer Engineering, Seokhwan Chung in Physics, and Keeyoung Moon in College of Library and Information Studies. I cannot say enough about my fiancé, Keeyoung. I have found my only consolation in her smile and hearty support.

Finally, I would like to thank my parents and my family for their everlasting trust and encouragement. Thanks Mom and Dad. Thanks my sisters, Eunsun and Eunyoung. Thanks my brother-in-law, Yongnam. Thanks my beloved nieces, Chunghwee and Yunsoo. Now, I am happy to dedicate my small achievement to them.

TABLE OF CONTENTS

Chapter 1: Introduction 1

1.1 The Problem 1
1.2 Semantic Regions 9
1.3 Summary 15
1.4 Scope 16
1.5 Content 17

Chapter 2: Related Work 18

2.1 Personal Information Storage 18
2.2 Information Organization and Retrieval 20
2.3 Flexible and Generalized Information Visualization 28
2.4 Summary 32

Chapter 3: Preliminary Works: Personal Media Management Tools 34

3.1 PhotoFinder3 34
3.2 Multimedia Bulletin Board 48
3.3 Summary 61

Chapter 4: MediaFinder: User Interface for Semantic Regions Construction and Operation 63

4.1 *Semantic Regions*: Interactive Personal Media Query Widget 63
4.2 MediaFinder 64
4.3 Constructing Semantic Regions in Media Finder 65
4.3.1 Semantic Region Construction 68
4.3.2 Semantic Specification 72
4.3.3 Model Management 76
4.4 Using Semantic Regions in MediaFinder 77
4.4.1 Data Binding (the *fling-and-flock* Metaphor) 78
4.4.2 Flexible Organization 81
4.4.3 Navigation 82
4.4.4 Meaning Extraction 83
4.4.5 Search 89

4.4.6 Indexing (Annotation) 94
4.4.7 Distribution 95
4.5 Summary 96

Chapter 5: MediaFinder Implementation 97

5.1 MediaFinder Architecture Overview 98
5.2 MediaFinder Code Overview 99
5.3 Data Management 104
5.3.1 Input File Format 104
5.3.2 Loading a data file 105
5.4 Graphical User Interface 108
5.4.1 Overall Structure of MediaFinder GUI components 108
5.4.2 Interaction Handlers 109
5.5 Query Processing 112
5.6 Performance 114
5.7 Extension 118

Chapter 6: Evaluating the Use and Construction of *Semantic Regions* 123

6.1 Evaluation of Semantic Regions Use 123
6.1.1 Procedure 124
6.1.2 Background Information 133
6.1.3 Result 134
6.2 Evaluation of Semantic Regions Construction 140
6.2.1 Procedure 141
6.2.2 Background Information 148
6.2.3 Results 150
6.3 Enhancement 161
6.3.1 Semantics Preview Control Panel 161
6.3.2 Enhancement of Semantic Regions Construction 164
6.4 Summary 166

Chapter 7: Conclusion 167

7.1 Contributions 167
7.2 Future Work 169
7.2.1 Re-implementation of MediaFinder 169
7.2.2 Scaling 170

7.2.3 Further evaluation ... 170
7.2.4 Extension to other types of data ... 171
7.3 Conclusions ... 172
References ... **173**
Appendix A: Data Structure ... **180**
A.1 SemanticRegion Class Structure ... 180
A.2 FileAttributeInfo Class used for specifying the semantics of the region ... 181
A.3 PersonalMediaItem Class representing a personal media item that has the form of a file ... 182
A.4 Semantic Regions Template file structure ... 182
Appendix B: Usability Study Survey ... **186**
B.1 Background Survey ... 186
B.2 Verbal Post Survey for First User Study ... 187
B.3 Verbal Post Survey for Second User Study ... 188
B.4 List of Measuring Criteria for the Second User Study ... 189

LIST OF TABLES

Table 5.5.1 The UpdateBooleanRegion function that updates all of the related Boolean regions starting from the argument region SR based on the priority hierarchy mechanism. 113

Table 6.1.1. Background information of 5 subjects participated in the first user study126

Table 6.1.2 The types and the amount of personal media data that pilot study users have. 134

Table 6.1.3 Task success (S: success, F: fail, SH: success with help) and time to completion (m: minute, s:second) 135

Table 6.1.4 User satisfaction survey of MediaFinder (1 – 9 scale) 139

Table 6.2.1 Background information of 3 experienced subjects for the second user study. The users' background information for two novice users was omitted because it was already done in the first user study (See **Table 6.1.1**) 143

Table 6.2.2 Success or failure of task (S: success, F: fail, SH: success with help) and time to complete (m: minute, s:second) 150

Table 6.2.3 User satisfaction survey on MediaFinder (1 – 9 scale) 159

LIST OF FIGURES

Figure 1.1.1 The file-folder hierarchy model for personal media organization 3

Figure 1.1.2 A typical folder hierarchy storage model, which has a wide and deep tree structure. The list of serial numbers in the selected folder, which are used as the names of the photo files, does not give users much information about the file. 5

Figure 1.1.3 Display representations examples (Continued)....................................... 7

Figure 1.2.1 Examples of *Semantic Regions*: 10

Figure 1.2.2 *MediaFinder* prototype implemented on top of PhotoFinder. Each region represents a person and contains all the photos annotated with the name defined in it. 11

Figure 1.2.3 Three mental models, friend groups, quarter year calendar, and US map (clockwise from the top "My Friend" group) are combined together. 12

Figure 1.2.4 (a) Select and apply the OR operation to the regions (b) New region created over the selected regions 13

Figure 2.2.1 The basic Lifestreams interface: All the documents are arranged based on the timeline. 21

Figure 2.2.2 Time-based personal information indexing and retrieval systems 22

Figure 2.2.3 SIS timeline visualization. The overview area at the left shows a timeline with hash marks representing the distribution of the search results over time. The highlighted region of the overview timeline corresponds to the segment of time displayed in the detailed view. To the left of the detailed timeline backbone, beyond basic dates, context is provided with landmarks drawn from news headlines, holidays, calendar appointments, and digital photographs. To the right of the backbone, details of individual search results (represented by icons and titles) are presented. 23

Figure 2.2.4 Visual information work spaces using spatial organization 24

Figure 2.2.5 ContactMap allows users to arrange their individual social networks in a visual map of individual contacts and groups: each node affords communication functions enabling users to retrieve current and archived information associated with them. 25

Figure 2.2.6 Presto Browser: In addition to a number of documents, the desktop contains four closed collections, two opened collections, and a set of predefined attributes. 26

Figure 2.2.7 Haystack is designed to let users manage all of their information in the way that makes the most sense to them. It allows users to define whichever arrangements of, connections between, and views of information they find most effective. 27

Figure 2.3.1 SpotFire for scientific and statistical data visualization 29

Figure 2.3.2 SageBrush. Brush's interface consists of a design work area (center) into which users drag prototypes (top), graphemes (left), and data names (bottom). . 30

Figure 2.3.3 Snap-together visualization: A coordinated-visualization interface for browsing census data. 31

Figure 2.3.4 The MVC abstraction diagram: Events typically cause a controller to change a model, or view, or both. Whenever a controller changes a model's data or properties, all dependent views are automatically updated. Similarly, whenever a controller changes a view, for example, by revealing areas that were previously hidden, the view gets data from the underlying model to refresh itself.32

Figure 3.1.1 PhotoFinder3 display with Library Viewer on the left, Collection Viewer with thumbnails on the upper right, and Photo Viewer on the lower right. Library Viewer shows a representative photo for each collection with collection information such as the collection name, description, date range, location, number of photos, and percentage of annotations and captions. Collection Viewer shows the photo thumbnails of the search result or the selected collection. Photo Viewer shows the real photo image of the selected thumbnail with annotations, captions, and other individual photo information. The list of names, showing all of the annotated people in the library, is used for search and direct annotation. 36

Figure 3.1.2 The schema of Photo Library database 37

Figure 3.1.3 Bulk annotation: multiple photos can be annotated with one drag and drop action just as direct annotation 40

Figure 3.1.4 PhotoFinder 4 extends direct annotation with user-defined category labels. In the left control panel, users can create their own category tags and use them to annotate photos. 41

Figure 3.1.5 Searching the annotated photos in PhotoFinder 44

Figure 3.1.6 Different displays in the Collection Viewer 46

Figure 3.1.7 When a mouse is over a thumbnail in CollectionViewer, a tooltip shows the information of the selected photo (query preview) and all the collections that contain the selected photo are highlighted red (brushing).................................. 47

Figure 3.2.1 Multimedia Bulletin Board message on the topic of "Visiting San Francisco".. 49

Figure 3.2.2 The example applications of spatial layout... 51

Figure 3.2.3 PhotoMesa: Spatial layout of photos using Quantum Treemaps 51

Figure 3.2.3 Meaning construction using spatial layout of information...................... 52

Figure 3.2.4 Example dynamic layout management through *DLT*............................. 59

Figure 3.2.5 (a) Personal web bookmarks with the original layout created by a single user. Similar to Data Mountain arrangement with occlusions............................ 60

Figure 4.3.1 Example models of *Semantic Regions* constructed with MediaFinder... 67

Figure 4.3.2 Creating, moving, and resizing *Semantic Regions* by Click-and-Drag... 68

Figure 4.3.3 Creating child regions inside a parent region... 69

Figure 4.3.4 Five basic views of a region: list, large icon, small icon, details, and thumbnails (clockwise from the top-left region) 70

Figure 4.3.5 Advanced views of *Semantic Regions*... 71

Figure 4.3.6 In the Semantic Regions Attributes dialog box, the file tab and the date tab are used for specifying the system-oriented file attributes for a region 73

Figure 4.3.7 Specifying the user-defined attributes for a region in the category and the people tab of the Semantic Regions Attribute dialog box. 74

Figure 4.3.8 The File summary properties in Microsoft Windows NT File System. File descriptions are used to store and retrieve the user-defined attributes such as category, people, and comments in MediaFinder. 76

Figure 4.3.9 Multiple models can be combined horizontally or vertically................. 77

Figure 4.4.1 The photos that do not satisfy the semantics of five state regions are collected in the "Remaining Items" region (red rectangle) located at the top of MediaFinder main window. 78

Figure 4.4.3 The maximized region shows more details about the contained personal media items or represents the hierarchy of child regions. 82

Figure 4.4.4 Region Brushing: an interaction technique to visualize the shared items among *Semantic Regions*... 85

Figure 4.4.5 Three models are combined horizontally to extract meanings about the interrelationships among the different models through Region Brushing........... 87

Figure 4.4.6 Extracting meanings concerning the interrelationships of the multiple models through the Region Brushing technique..88
Figure 4.4.7 Applying the Boolean AND operation to the multiple regions..............90
Figure 4.4.8 The steps for applying Boolean operations to get the Boolean region of (NOT (Catherine Plaisant)) AND ((Ben shneiderman) OR (Jenny Preece))) 92
Figure 4.4.9 Indexing the personal media items by dragging them onto the *Semantic Region*..95
Figure 5.1.1 MediaFinder software architecture...99
Figure 5.4.1 A schematic overview of the container classes used in the MediaFinder GUI. The entire window is an instance of frmSemanticRegion – an inherited System.Windows.Forms.Form class...108
Figure 5.7.1 Average times for MediaFinder to completely process queries – including search and display update – on several query types. Results are shown for data sets of 100, 500, 1000 and 3000 items with 4 different query tests. (1) Constructing a *Semantic Region* by dragging a metadata tag from the Semantics Preview Control Panel (2) Dragging the personal media items onto a built-in model to measure the performance of the *fling-and-flock* metaphor (3) Constructing a Boolean region by applying a UNION operation to two regions (4) Highlighting the shared items among regions by placing the mouse over the title of a region...117
Figure 5.7.2 The Extension of MediaFinder to database table. In this example, five different attributes (adequacy of care, education of mother, race of mother, race of father, and marital status of mother attributes) were joined together to visualize the interelationships between the attributes. All the records representing unmarried mother were highlighted through the Region Brushing feature to visualize the distributions of unmarried mother with respect to four other attributes. 121
Figure 6.1.1 User study data set: a total of 281 photos chosen from the CHI photo library (from CHI91 to CHI2000). Metadata about location, conference name, date, and people have been combined to photo files using PhotoFinder4..........130
Figure 6.1.2 US map mental model with 7 state Semantic Regions and 8 cities of child regions. 130
Figure 6.1.3 Conference calendar mental model: each region represents a CHI conference between CHI91 and CHI2000...131

Figure 6.1.4 People mental model: the 9 most frequently appearing people from the test data set were chosen. 131

Figure 6.1.5 Combined-model (People, calendar, and location) 132

Figure 6.2.1 US map mental model with 5 state regions and 5 child regions of cities. Users can specify location semantics for a region by selecting a tag from the location category in the Semantics dialogbox 145

Figure 6.2.2 A people model consists of three Semantic Regions. Users can choose a person from the list of people in the semantic dialog box. The list of people was collected from the metadata of bound photos. 146

Figure 6.2.3 (a) People and location model combined horizontally (b) People model imported inside Location model (c) Location model imported inside People model 147

Figure 6.2.4 Screenshots of the models created by the subjects in task 4 157

Figure 6.3.1 The MediaFinder interface consists of two separate windows: the Semantics Preview Control Panel (left window) and the Main Panel (right window). The Semantics Preview Control Panel is composed of four tab windows to show all the metadata collected from the bound personal media items. The Main Panel is composed of two tab windows to visualize the personal media items either in spatially organized form or in linear listed form. 162

Figure 6.3.2 Four different views of the collected metadata shown in the Semantics Preview Control Panel. 163

Figure 6.3.3 Two different views of the main panel: 164

Figure 6.3.4 Simplified region construction and semantics specification with one simple drag and drop action. 165

Chapter 1:

Introduction

1.1 The Problem

Computer users deal with large numbers of personal media objects such as images, audio clips, voice mail, video clips, web pages, emails, and various document files. This vast array of personal media objects presents specific challenges for everyday users. Users often struggle to interpret, explore, arrange, and use personal media objects because of three major problems; an ever increasing amount of personal media data, rigid organizing metaphor, and difficulty in rapid data access.

After the development of the microprocessor in 1971, the computer has evolved at the annual rate of 1.60 (a doubling every 18 months, or 100 times per decade) known as Moore's Law [41]. Magnetic density and fiber optic data transmission rates have also evolved at the rate of Moore's law. According to this law, personal computers will be capable of storing a terabyte (1000 gigabytes) of information on an individual machine at a very low cost within a few years [9]. In addition, there has been extraordinary progress in all of the technologies for so-called "cyberization" [8] ranging from processor speed, storage capacity, scanner speed, audio/video encoding, camera and display resolution and so on. With the progress of computer hardware and cyberization technologies, users began to create, acquire, and manage more and more personal media on their individual machines.

Moreover, based on the progress of the technologies, there have been a few research attempts to encode and store all of one's personal information [10][37][40][67] for personal and professional use. In the CyberAll [8] project, the archive includes books, CDs, correspondence (such as letters, memos, and email), transactions, papers, photos and albums, and video. Their study shows that users will be able to record all of the information accumulated in their entire personal and professional life in a few terabytes, including everything spoken and written, but not everything captured via video. Hence, it does not seem to be an unrealizable dream to encode and store all of a person's information on his personal machine within the next decade.

By keeping every bit of information, users will be able to get the answers for any type of question associated with this corpus such as:

- Recalling the Paris hotels that I have been in during the last ten years.
- Finding the video clips of my friend's wedding in California about five years ago.
- Finding all the slides, audio clips, and photos about the speeches and lectures I did outside the US during 2000
- Showing the photos from a trip to Spain. Or, taken during July 1999.
- Finding the articles, papers, letters, emails, etc. that mention SIGCHI.
- Finding the music file of Beethoven's Sonata No.14 in C#-, Op.27 No.2 'Moonlight'

In summary, it is clear that users will be exposed to a deluge of personal media in the near future and it will be a serious challenge for users to manage the enormous volume of personal media efficiently on their personal computers. Personal media management includes organization, meaning extraction, search, navigation, indexing, and distribution.

Figure 1.1.1 shows the most common file folder hierarchy model for information organization. However, the use of folders as a fundamental organizing principle, and the restriction that personal media appear in only one folder at a time, forces users to create strict categorizations. This results in inflexible organizations that tend to persist over time even though their needs evolve [46].

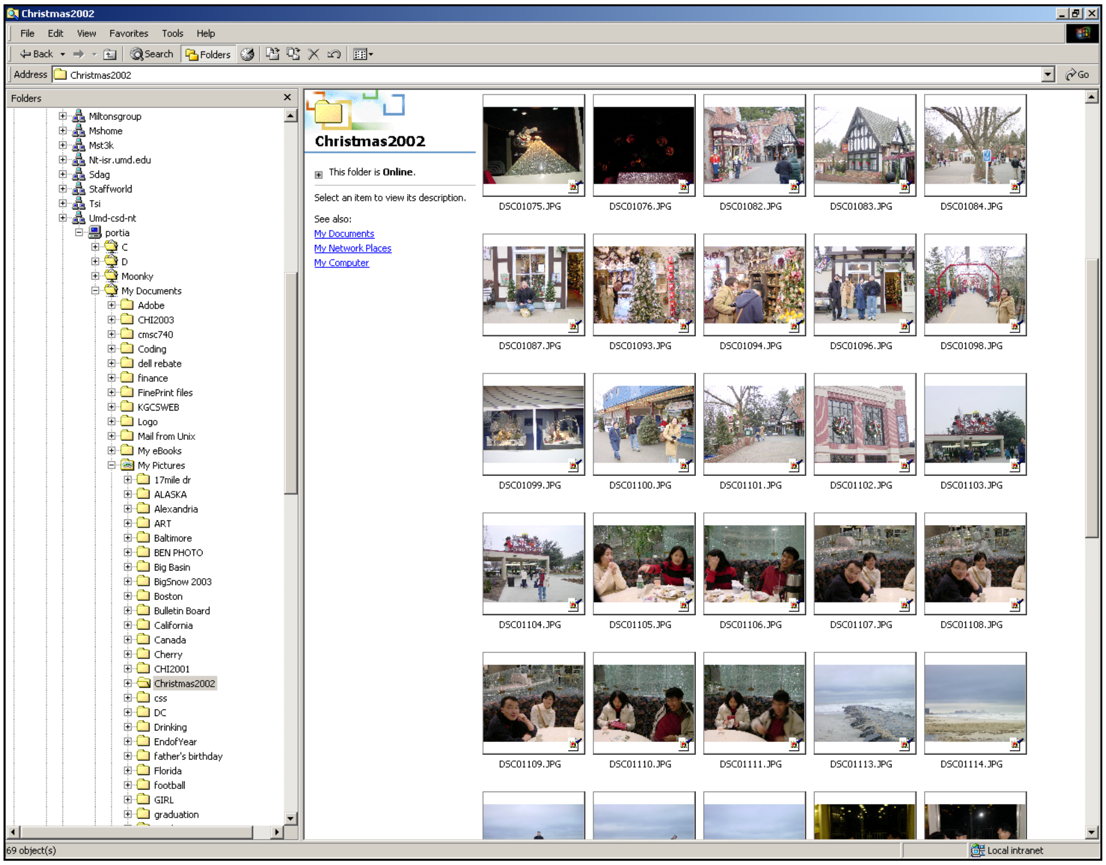


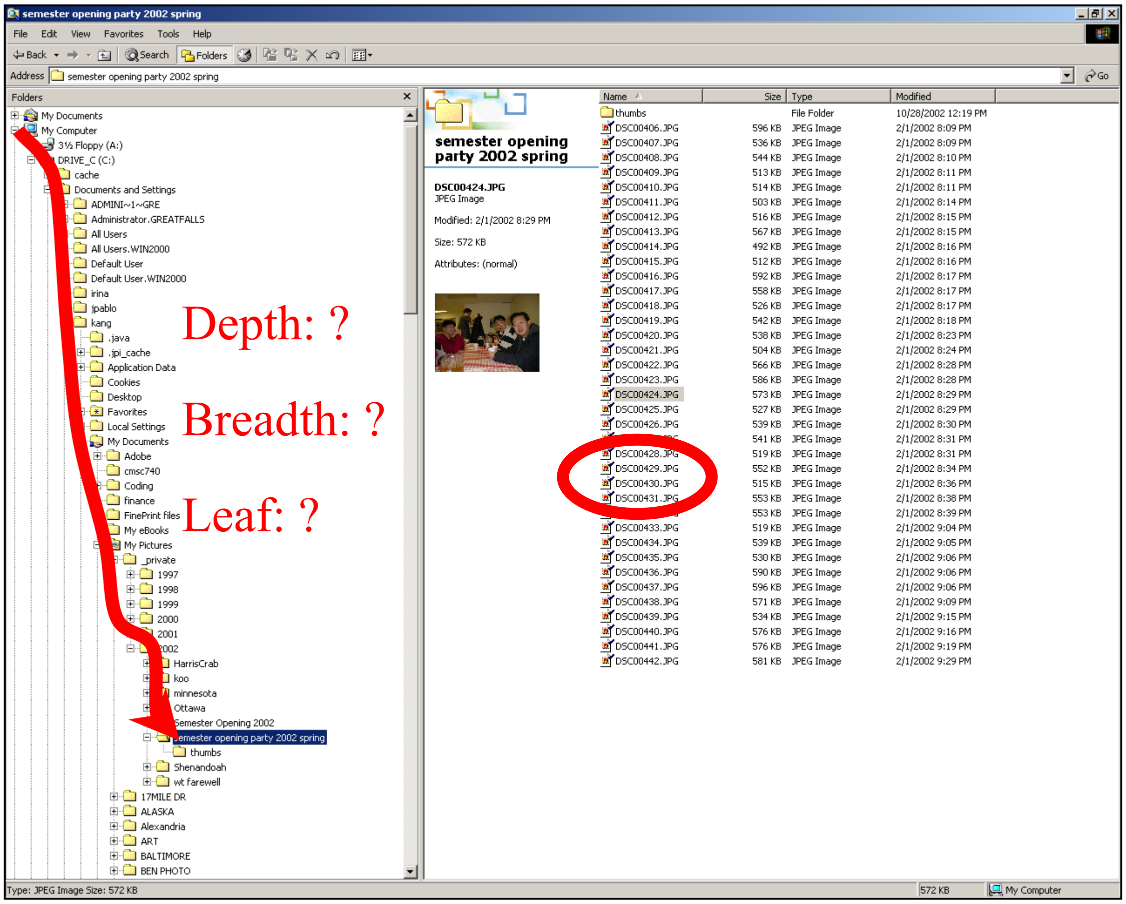
Figure 1.1.1 The file-folder hierarchy model for personal media organization

Users usually understand the data by constructing models of it in their minds. There is no unique or right model. Rather, the mental model is very personal, has meaning for the individual who creates it, and is tied to a specific task. For a simple example of personal photo library, photo collections can be organized by timelines, locations, events, people, etc. depending on users' mental models. Despite the diversity of users' mental models, the means available for users to organize and customize their information spaces are extremely poor and driven mostly by storage and distribution models, not by users' needs. Consequently, one of the main challenges in designing and implementing a novel user interface for exploring personal media is to make it easier for end-users to construct and represent their mental models and to apply them to personal media data for more flexible and task-specific personal media management.

There have been some research efforts made to support more flexible document management against the file system approach. The LifeStreams [21] system uses a timeline as the major organizational metaphor for managing document spaces. Lifestream is motivated by the problems of a standard single-inheritance file hierarchy, but simply replaces one superordinate aspect of the document (its location in the hierarchy) with another (its location in the timeline).

The Semantic File System [24] introduces the concept of a "virtual directory". In this system, virtual directory names are interpreted as queries and thus provide flexible associative access to files and directories in a manner compatible with existing file systems. However, the system is directly integrated into a file system to enrich command line programming interfaces and no graphical user interface features are introduced other than the filename/query language syntax.

Presto [20] is a prototype document management system providing rich interaction with documents through user-level document attributes. In this model, users interact with their documents in terms of high-level attributes that are provided and fluidly managed by the users themselves. However, the high-level document attributes are mainly used for searches but not for other document management tasks such as organization and meaning extraction. In addition, the design of 2D visual query interface on which all the documents, collections, and attributes are scattered often prevents users from constructing their mental models concerning the documents.



DSC00424.JPG
JPEG Image

Modified: 2/1/2002 8:29 PM

Size: 572 KB

Attributes: (normal)

Name	Size	Type	Modified
thumbs		File Folder	10/28/2002 12:19 PM
DSC00406.JPG	596 KB	JPEG Image	2/1/2002 8:09 PM
DSC00407.JPG	536 KB	JPEG Image	2/1/2002 8:09 PM
DSC00408.JPG	544 KB	JPEG Image	2/1/2002 8:10 PM
DSC00409.JPG	513 KB	JPEG Image	2/1/2002 8:11 PM
DSC00410.JPG	514 KB	JPEG Image	2/1/2002 8:11 PM
DSC00411.JPG	503 KB	JPEG Image	2/1/2002 8:14 PM
DSC00412.JPG	516 KB	JPEG Image	2/1/2002 8:15 PM
DSC00413.JPG	567 KB	JPEG Image	2/1/2002 8:15 PM
DSC00414.JPG	492 KB	JPEG Image	2/1/2002 8:16 PM
DSC00415.JPG	512 KB	JPEG Image	2/1/2002 8:16 PM
DSC00416.JPG	592 KB	JPEG Image	2/1/2002 8:17 PM
DSC00417.JPG	558 KB	JPEG Image	2/1/2002 8:17 PM
DSC00418.JPG	526 KB	JPEG Image	2/1/2002 8:17 PM
DSC00419.JPG	542 KB	JPEG Image	2/1/2002 8:18 PM
DSC00420.JPG	538 KB	JPEG Image	2/1/2002 8:23 PM
DSC00421.JPG	504 KB	JPEG Image	2/1/2002 8:24 PM
DSC00422.JPG	566 KB	JPEG Image	2/1/2002 8:28 PM
DSC00423.JPG	586 KB	JPEG Image	2/1/2002 8:28 PM
DSC00424.JPG	573 KB	JPEG Image	2/1/2002 8:29 PM
DSC00425.JPG	527 KB	JPEG Image	2/1/2002 8:29 PM
DSC00426.JPG	539 KB	JPEG Image	2/1/2002 8:30 PM
[illegible]	541 KB	JPEG Image	2/1/2002 8:31 PM
DSC00428.JPG	519 KB	JPEG Image	2/1/2002 8:31 PM
DSC00429.JPG	552 KB	JPEG Image	2/1/2002 8:34 PM
DSC00430.JPG	515 KB	JPEG Image	2/1/2002 8:36 PM
DSC00431.JPG	553 KB	JPEG Image	2/1/2002 8:38 PM
[illegible]	553 KB	JPEG Image	2/1/2002 8:39 PM
DSC00433.JPG	519 KB	JPEG Image	2/1/2002 9:04 PM
DSC00434.JPG	539 KB	JPEG Image	2/1/2002 9:05 PM
DSC00435.JPG	530 KB	JPEG Image	2/1/2002 9:06 PM
DSC00436.JPG	590 KB	JPEG Image	2/1/2002 9:06 PM
DSC00437.JPG	596 KB	JPEG Image	2/1/2002 9:06 PM
DSC00438.JPG	571 KB	JPEG Image	2/1/2002 9:09 PM
DSC00439.JPG	534 KB	JPEG Image	2/1/2002 9:15 PM
DSC00440.JPG	576 KB	JPEG Image	2/1/2002 9:16 PM
DSC00441.JPG	576 KB	JPEG Image	2/1/2002 9:19 PM
DSC00442.JPG	581 KB	JPEG Image	2/1/2002 9:29 PM

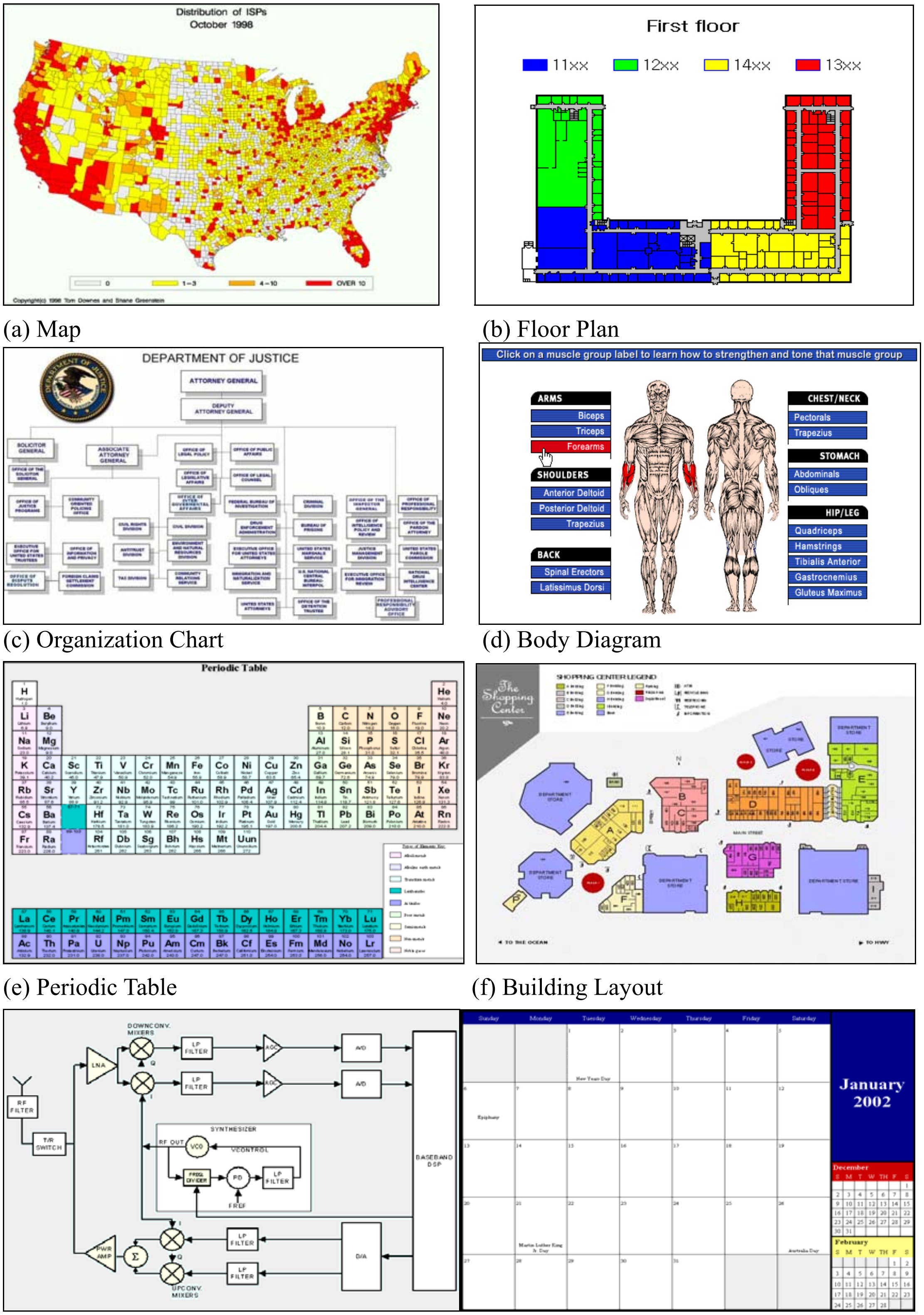
Type: JPEG Image Size: 572 KB

572 KB

Figure 1.1.2 A typical folder hierarchy storage model, which has a wide and deep tree structure. The list of serial numbers in the selected folder, which are used as the names of the photo files, does not give users much information about the file.

Another challenge in designing a personal media explorer is to find ways to enable users to access media objects as fast as possible. The wide and deep structure of file folder hierarchies as well as inconsistent and meaningless file names often forbid users to rapid data access from storage (**Figure 1.1.2**). However, human capabilities for spatial cognition can be a solution for this problem because they often allow rapid information retrieval in 2D graphical user interfaces. Several prior experiments [54][18] have shown that spatial organizations of information enable users to access data items surprisingly quickly. Furthermore, if the spatial organization of information is based on the semantics of well-known display representations (e.g. map, calendar, organizational chart, timeline, genealogy tree, and so on) (**Figure 1.1.3**) or users' preferred mental models, it is expected to improve task performance even more.

Numerous display representations (e.g. diagrams and charts) are used for showing information because they often make data more comprehensible, ensure faster data perception, and give clearer intuition for data ontology. **Figure 1.1.3** shows some examples of familiar display representations used for displaying information, which include calendars, timelines, maps, organization charts, genealogy trees, floor plans, body diagrams, flow charts, building layouts, integrated circuit block diagrams, tournament charts, periodic tables, and so on. Unlike the standard mathematical or statistical visualizations such as histogram, line graph, pie chart, scatter plot, list, table, grid, etc., most of the display representations in these examples contain inherently unique semantic information in their visual patterns. For example, organizational charts and genealogy trees include semantic information on the hierarchical structures of groups as well as their relationship. Timelines and calendars represent temporal information concerning events as well as time units.



(a) Map

(b) Floor Plan

(c) Organization Chart

(d) Body Diagram

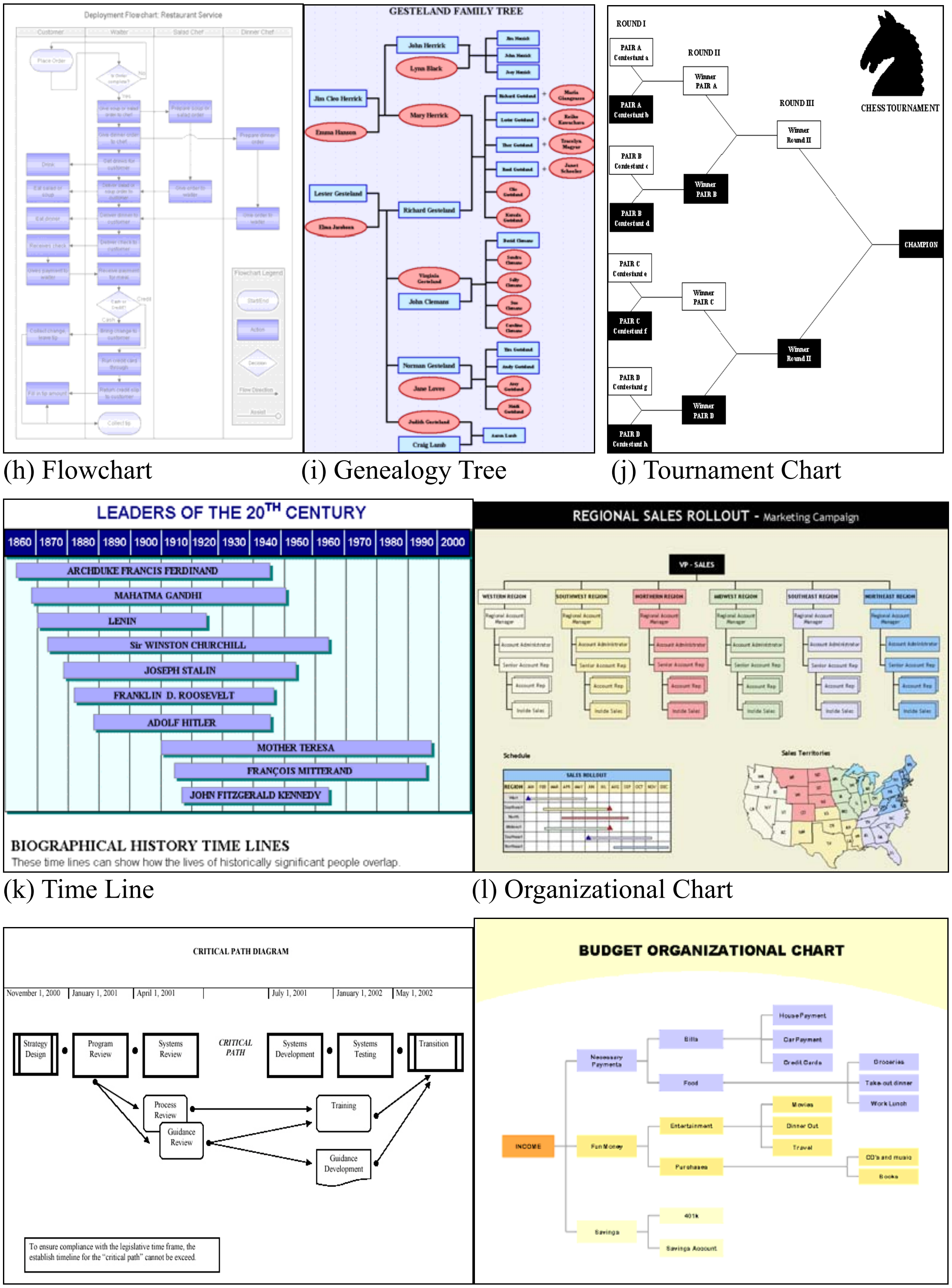
(e) Periodic Table

(f) Building Layout

(g) Integrated Circuit Block Diagram

(g) Calendar

Figure 1.1.3 Display representations examples (Continued)



(h) Flowchart (i) Genealogy Tree (j) Tournament Chart

(k) Time Line (l) Organizational Chart

(m) Critical Path Chart (n) Organizational Chart

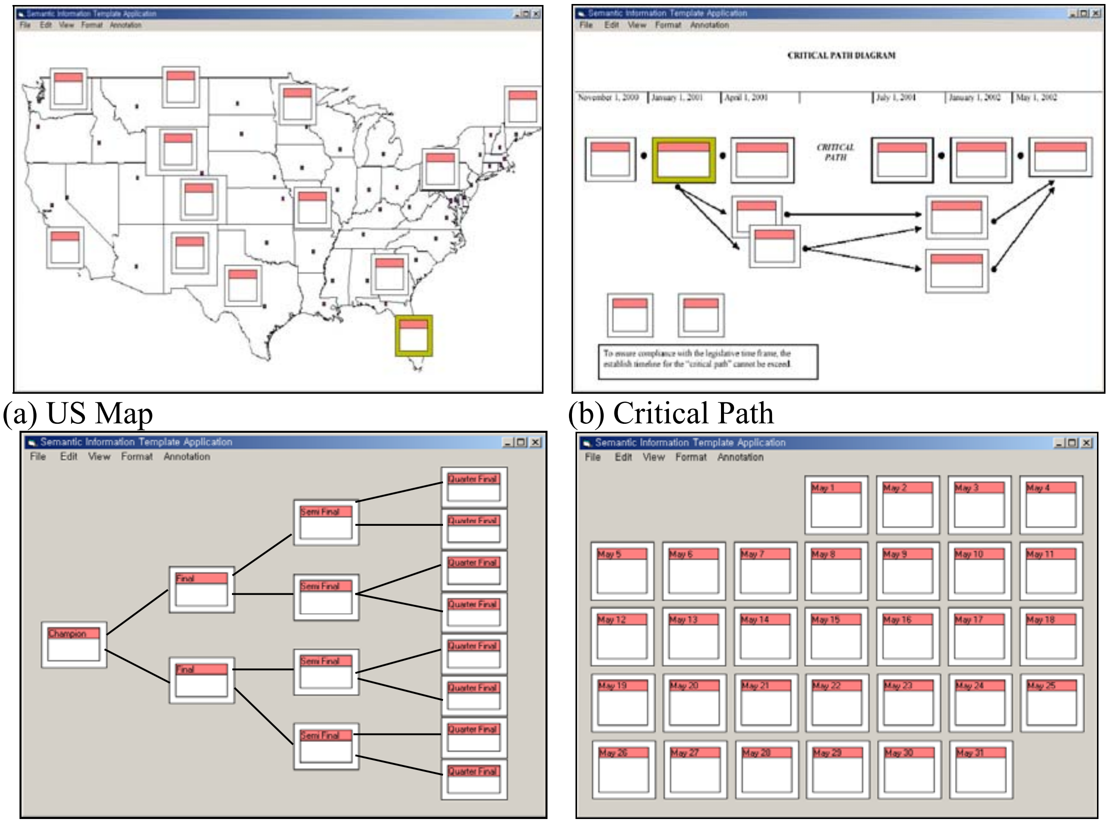
Figure 1.1.3 Display representations examples

The semantic combination of display representations and personal media can provide multiple perspectives for different aspects of an information task. For example, suppose that a huge number of digital photos are combined with various display representations. The photos can be combined with a map display (**Figure 1.1.3**(a)) to show the distribution of photos according to the place they were taken. They can also be combined with a calendar display (**Figure 1.1.3**(g)) to represent the distribution of photos over days, weeks, months, or years depending on the time unit of the calendar. If the photos are about the anatomy of the human body, they can be combined with a human body diagram (**Figure 1.1.3**(d)) to specify which picture comes from which part of the human body. If the photos are about paintings taken from a museum, they can be combined with the museum layout diagram (**Figure 1.1.3**(f)) to indicate which painting was displayed in which exhibition room in the museum, or they can be combined with a timeline of art history (**Figure 1.1.3**(k)) to identify the trends of paintings at the time they were created. If the photos are about the basketball games of the NCAA tournament, they can be combined with the tournament chart (**Figure 1.1.3**(j)) to specify which photo comes from which tournament game in the chart.

1.2 Semantic Regions

To tackle the problems mentioned in the previous section, this dissertation presents an innovative approach, *Semantic Regions,* based on the hypothesis that "Spatial organization of information based on the semantics of personal media" and the "Use of various users' mental models for managing the personal media" will greatly improve task performance and user satisfaction. A prototype of the personal media management and exploring application, MediaFinder*,* which is based on the concept of *Semantic Regions,* is also presented.

Semantic Regions are rectangular regions drawn directly on a two-dimensional space. Users can specify the shapes, sizes, and positions of the regions in two-dimensional space and thus form the layout of the regions meaningful to them (**Figure 1.2.1**). Creation of *Semantic Regions* is straightforward: users simply click on the desired starting point of the semantic region and drag the pointer to the desired location of the opposite corner. Once the semantic region is created, the semantics of each region can be specified with the combination of the attributes of personal media items. The child *Semantic Regions* can also be defined recursively inside each region.



To ensure compliance with the legislative time frame, the establish timeline for the "critical path" cannot be exceed.

(a) US Map (b) Critical Path

(c) Organization Chart (d) Monthly Calendar

Figure 1.2.1 Examples of *Semantic Regions*:

When users drag a collection of the personal media items onto the S*emantic Regions*, the items are automatically distributed and placed in the appropriate regions based on the semantics defined in each region. This metaphor is called *fling-and-flock*.

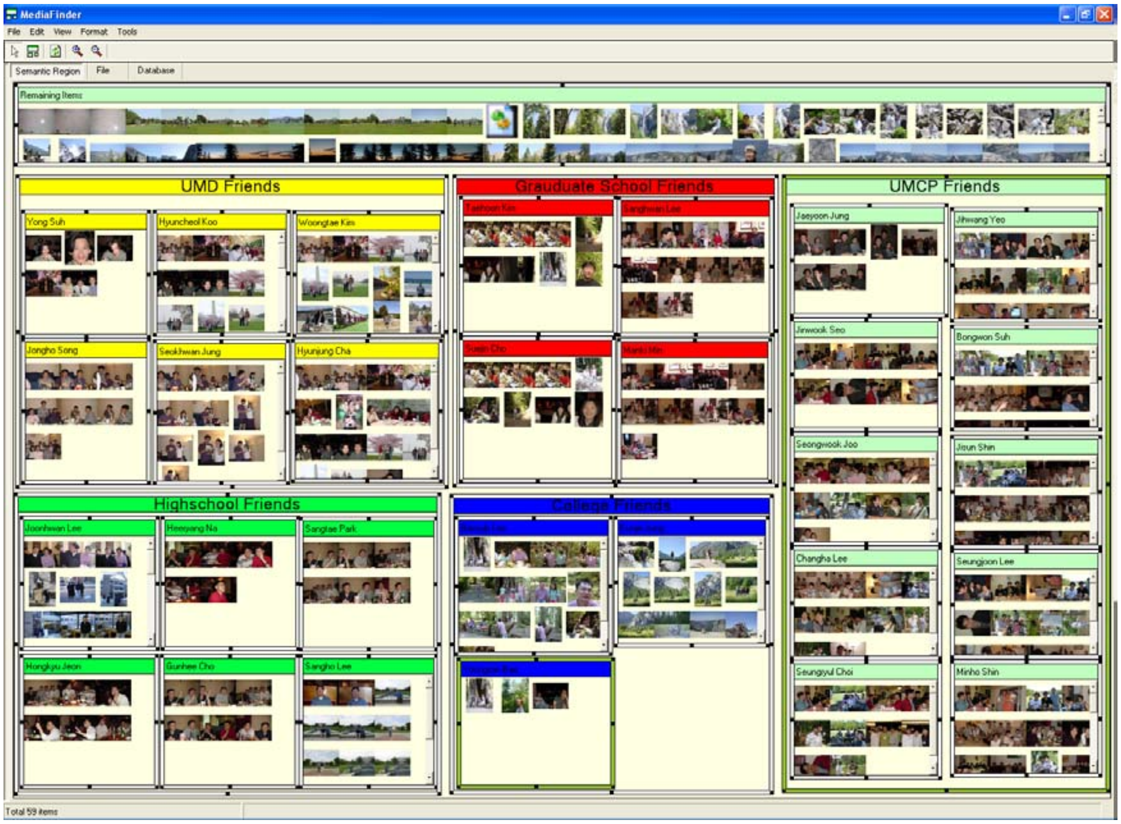


Figure 1.2.2 *MediaFinder* prototype implemented on top of PhotoFinder. Each region represents a person and contains all the photos annotated with the name defined in it. The regions are grouped into 5 clusters to represent different friend groups (UMD friends, highschool friends, graduate school friends, college friends, and UMCP friends). Each group has its own color to represent the different group of friends.

The *MediaFinder* is a prototype interactive tool built to investigate the use of *Semantic Regions* for exploring and managing personal media data. The initial prototype has been built on top of PhotoFinder [61] to manage photos. In *MediaFinder,* users create and arrange the rectangular regions on 2D space and specify the semantics. In **Figure 1.2.2**, each region represents a person, and the regions are grouped into 5 clusters to represent different friend groups. When photos are dragged onto the regions, they are automatically placed in the appropriate regions based on the annotations. This metaphor is called *fling-and-flock*; that is, users fling the objects and the objects flock to the regions. If photos do not satisfy any of the semantics of regions, they are collected in the remaining items region located at the top of the panel. Users can save the current state of regions and subsequently apply it to other personal media data sets.

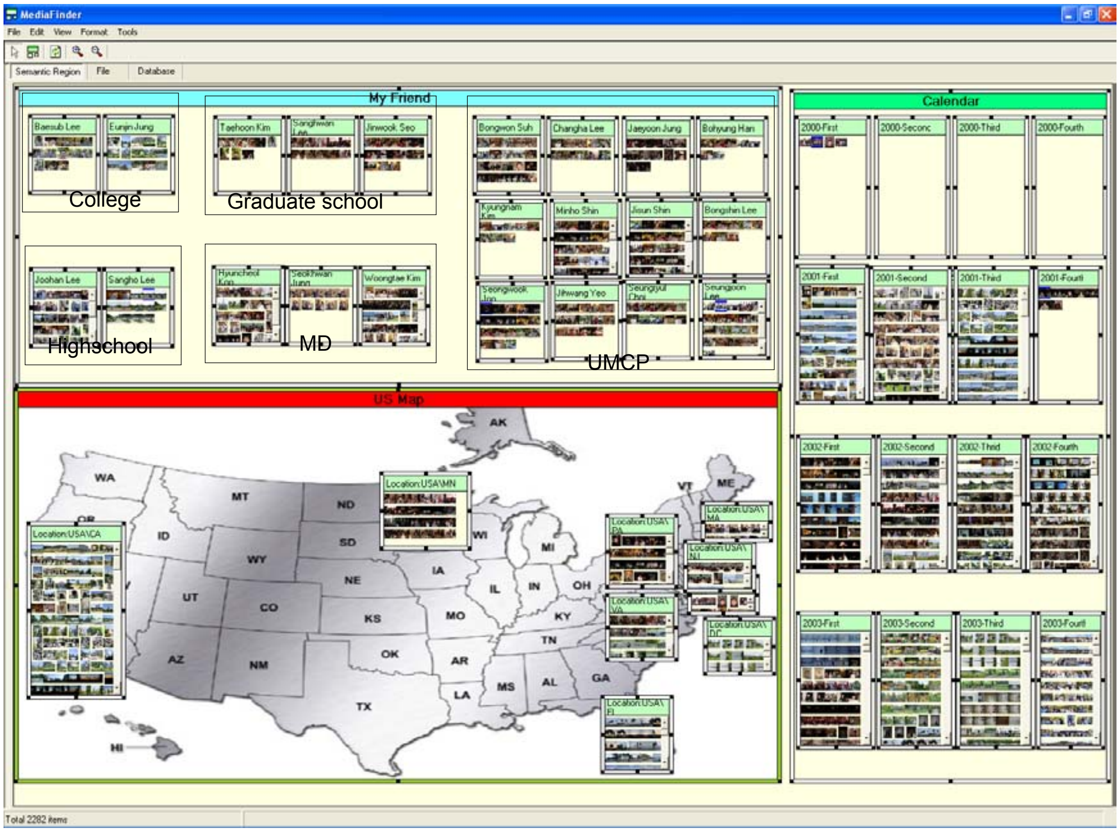


Figure 1.2.3 Three mental models, friend groups, quarter year calendar, and US map (clockwise from the top region) are combined together. The regions in the My Friend region are grouped into 5 clusters to represent different friends groups (high school friends, college friends, graduate school friends, UMD CS friends, and MD friends). Each region in the Calendar group represents a quarter year from 2000 to 2003.

Users can combine a few heterogeneous mental models simultaneously to observe the relationships among them. In **Figure 1.2.3**, three mental models, the US map, the year calendar, and my friends group are combined together and bound to photos through the *fling-and-flock* metaphor.

A photo can be contained in multiple regions as long as it satisfies the semantics of regions, which enables flexible organization. When the mouse is over the title of the region (yellow circle), all the regions containing any of the items in the region are highlighted red. In this example, users can easily answer questions like, “When did I visit California and whom did I meet there ? ” by positioning the mouse on the title of California region.

MediaFinder supports Boolean operations (AND, OR, NOT) of the regions to facilitate search and navigation. As indicated in **Figure 1.2.4**, to see all the photos taken in the first quarter of years between 1997 and 2002, the first column of each row is selected to apply the OR operation. A new region is created over the selected regions to contain the photos in this group.

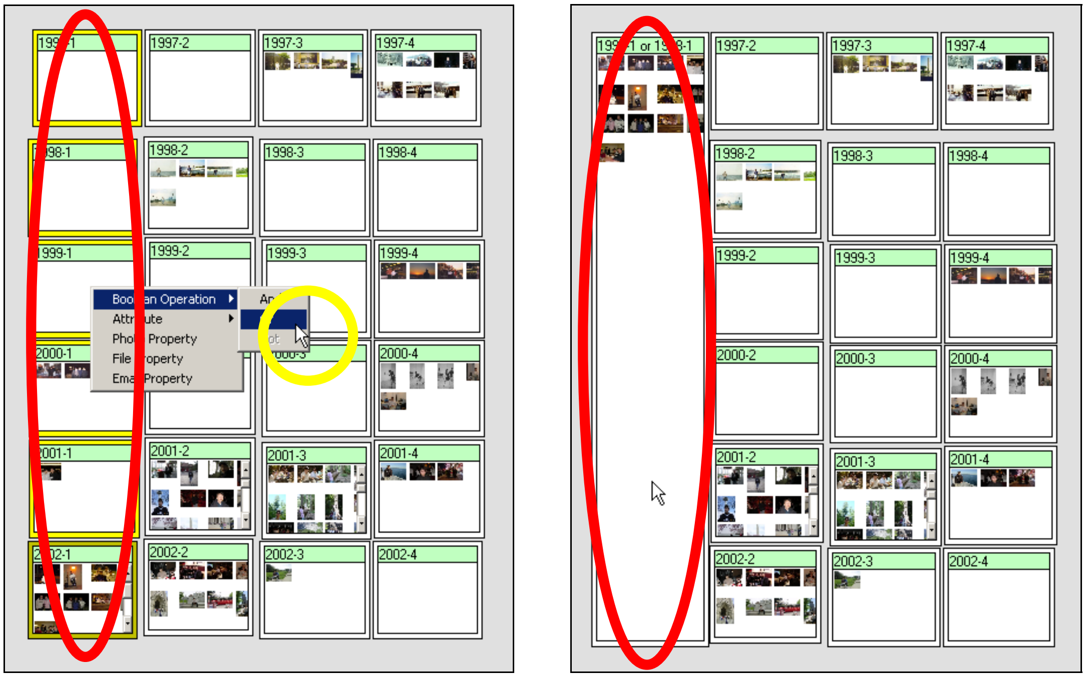


Figure 1.2.4 (a) Select and apply the OR operation to the regions (b) New region created over the selected regions

Semantic Regions can also be used for indexing personal media items. If users select personal media items in one region and drag them to another region, the attribute values of the selected items are updated with those specified in the dragged region.

In this section, the basic concept as well as the usage of *Semantic Regions* has been introduced to present a quick overview of how this approach can be applied to the personal media management tasks such as organization, meaning extraction, search, navigation, indexing, and distribution. The main focus of this research is to help wider range of users easily construct their various mental models for their personal media data and represent them spatially on two dimensional space to improve the performance of personal media management tasks as well as user satisfaction.

Users usually understand personal media data by constructing their own mental models in their minds. It is interesting to note that users have different mental models even for the same data set. Moreover, they organize their mental maps differently even if they have similar mental models. For an example of music, people use different classifications of music depending on their interests and tastes. Some people may classify music into three general categories such as pop, classical, and Christian, but others may break down into more details like rock, jazz, R&B, rap, country, classical, Latin, Christian, and so on. Even in the same classical category, some may categorize music by historical periods such as Renaissance, Baroque, Classical, Romantic, Modern, on the other hand, others may classify music into several forms and genres such as ballads, canons, concertos, etudes, fantasies, fugues, inventions, oratorio, requiems, sonatas, symphonies, variations, and so on.

Although they have similar categorizations of music, their mental maps can be differently organized and represented on 2D space. For example, some people may see

R&B as closer to rock, whereas others see blues as closer to country and to jazz. Furthermore, they may make use of size to show the strength of their interests or number of items they have (e.g. rock might be twice as large as R&B, and four times as large as jazz). They may even make use of shapes or connecting lines to make their mental maps to be more understandable and recognizable.

For another example, people may have a rough map of their neighbors. Unlike an actual map, it can represent a personal cognitive map of strength of interest in a local park, a neighbor's house, a grocery store, a public swimming pool, library, parking lot, etc. Although, the map may be distorted and have a few different scales depending on their strength of interest or number of personal media items related to the places in the map, it helps them understand, recognize, and explore their information on their own way.

As shown in the above examples, there is no right or unique mental model for the personal media data. Because of this diversity of models, it is not easy for users to manage their personal media data with a fixed and rigid organizing metaphor of current file management system. The *Semantic Regions* approach tries to solve this problem by providing users an environment to customize their information space appropriate for their mental models and specific tasks. The spatial organization of information based on users' personal mental maps reinforces this approach.

1.3 Summary

The concept of *Semantic Regions* introduces an innovative way to construct users' mental models by drawing regions on the 2D virtual space and applying various personal ontologies[28] to the personal media items using the *fling-and-flock* metaphor.

A prototype personal media management application, *MediaFinder*, verifies the usability of the interface, particularly in comparison with alternative approaches. Empirical studies help clarify the benefits and drawbacks of *MediaFinder*, while suggesting additional interface improvements.

The current research on *Semantic Regions* contributes a few major innovations. It extends the concept of the existing system-driven file management system to the user-oriented personal media management system by employing users' mental models toward the data. The *fling-and-flock* metaphor as well as the concept of *Semantic Regions* has applicability for visual organization of many data domains including scientific and statistical data. Also, this research shows a novel interactive environment for constructing and visualizing queries for personal media data. Finally, the visual design, dynamic aspects, and user controls implemented in *Semantic Regions* are among its many contributions.

1.4 Scope

MediaFinder mainly focuses on organization, meaning extraction, search and navigation of personal media using the semantics defined in personal media items. MediaFinder does not address other classes of personal media management tasks such as indexing or distribution.

Although MediaFinder supports a manual annotation mechanism, it is basically assumed that the metadata (attribute values) of personal media items is predefined. This assumption is based on the fact that a variety of metadata of personal media data will be available in the near future through various new technologies such as automatic online creation (e.g. email and file attributes or digital photo metadata), automatic offline extraction (e.g. face recognition, keyword extraction, video indexing, etc.), accessing other sources (e.g. freeDB and imDB), manual creation with advanced tools (e.g. PhotoFinder), and so on.

MediaFinder focuses on common types of personal media data typically encountered in the daily life of general computer users, such as email, web pages, photographs, movie clips, music files, ebooks, and various office documents. This research is less concerned with scientific data and visualization even though it is extensible.

1.5 Content

Chapter 2 reviews related literature and Chapter 3 describes the preliminary works of PhotoFinder and Multimedia Bulletin Board as a motivation of this research. Multiple views built in PhotoFinder as well as dynamic layout management in Multimedia Bulletin Board are introduced to form the basis of a preliminary formal model of *Semantic Regions*. Chapter 4 describes the MediaFinder user interface for *Semantic Regions* construction and operation. Chapter 5 describes MediaFinder's architecture and data structures. Chapter 6 details two usability studies on the construction and operation of *Semantic Regions*. Chapter 7 concludes with benefits, limitations, contributions, and future work.

Chapter 2:

Related Work

The focus of this dissertation is personal media data. There is no formal definition of the term “personal media”, but people both in research and in industry usually use this term to represent all of the computerized files and data that users handle on their personal computers. It includes images, audio clips, music files, moving pictures, emails, web pages, and various document files and sometimes even represents entire computer file systems.

There has been a great deal of research related to personal media, and they can be collectively classified into three general areas: storage, retrieval, and visualization. Among them, three specific areas are directly related to the topics addressed in this dissertation. They are: “Personal Information Storage”, “Information Organization and Retrieval”, and “Flexible and Generalized Information Visualization”.

2.1 Personal Information Storage

With the extraordinary progress made in “cyberization”[8] technologies, ranging from processor speed, storage capacity, data transmission rate, scanner speed, camera resolution, audio/video encoding, printing/display quality, etc., there have been a great deal of research efforts made concerning Personal Information Storage.

The Factoid project [40] at Compaq Western Research Lab imagined small ubiquitous devices that would capture many small facts (such as one might see on a sign, on a business card, in an advertisement, or on the display of an instrument like a thermometer or GPS receiver) per day and which are each only a few hundred bytes in size. Based on 1000 facts a day, they calculated 73MB/year (highly compressible) which can thus "be saved forever and constitute a sort of history of the users' life".

In the CyberAll [8] project at Microsoft, G. Bell tries to archive all of his personal and professional information content including that which has been computer generated, scanned and recognized, and recorded on VHS tapes. The archive includes books, CDs, correspondence (such as letters, memos, and email), transactions, papers, photos and albums, and video taped lectures. Their study estimates that a user will be able to record all of the information accumulated in their entire personal and professional life in a few terabytes, including everything spoken and written, but not everything captured via video. Their study has also reported issues surrounding this project such as Longevity and Long-Term Retrievability, Access and Access Control, Database and Retrieval Tools for Non-Textual Information, and Usability.

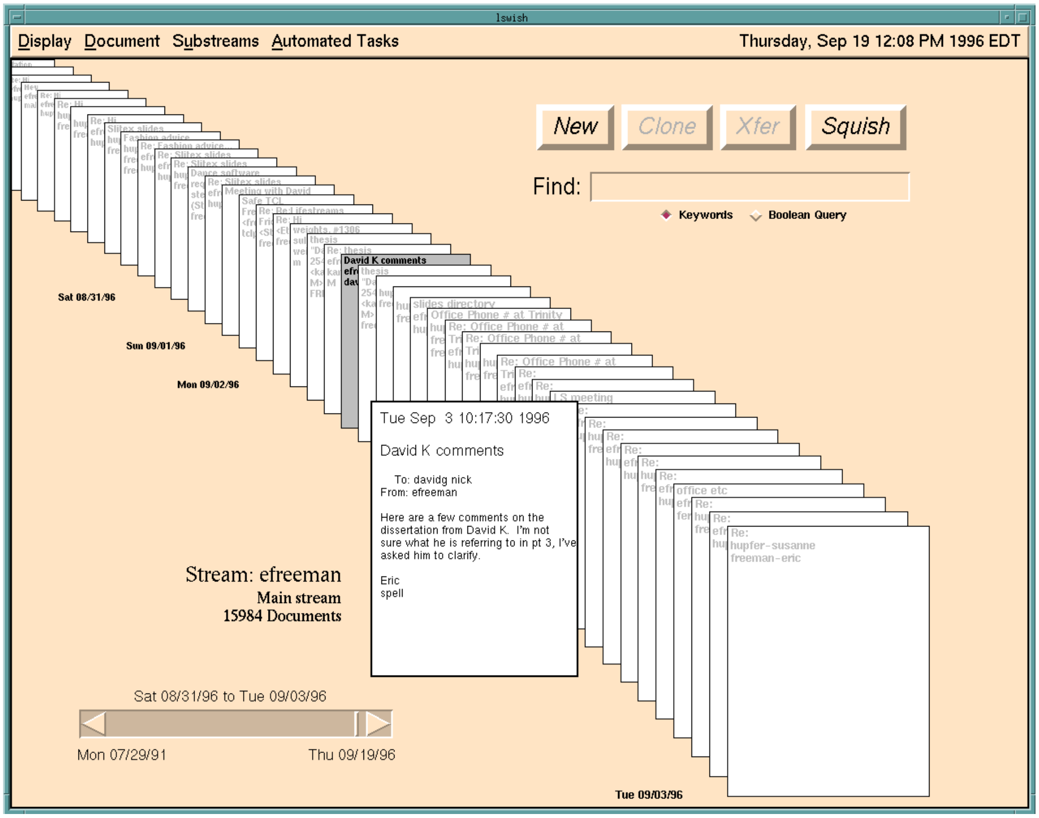
There have been a few projects at XRCE (Xerox Research Center Europe) Cambridge on augmented memory. "Pepys"[45] used automatically captured location information from active badges to yield descriptions of past activities. It then combined these data into daily diaries of a user's activities. The main idea was that by providing these diaries, retrieval of information about the activities occurring over the course of the day would be improved. "Video Diary"[73] enhanced the diary information presented by *Pepys* by providing automatically-captured video images of the activities of the user. This information was then presented as a video diary of a person’s activities over the course of a day. In “Forget-Me-Not”[37], information is automatically collected about a user's activities such as telephone calls made, e-mail notes received, and meetings attended. This information is then stored and presented as a personal biography with activities presented in time order. The biography can be browsed, and it may be filtered to show only a particular type of activity.

In the “I Sensed Series” project [17], Brian Clarkson at MIT has built a wearable computer that records video (including looking backwards), audio, and orientation data for up to 10 hours a day. A 10GB disk is used to record up to 2 days data. The purpose of this project is to explore how to statistically represent the experiences of a wearable computer user for the purposes of day-to-day behavior prediction and classification.

2.2 Information Organization and Retrieval

As shown in the previous section, due to the progress in technology and a corresponding desire by users to digitize and keep their personal information on their machines, the amount of personal media data stored on individual machines will be increasing exponentially. Consequently, indexing, retrieval, and management of personal media become critical issues. Several research attempts related to document management try to resolve similar problems by supporting a more flexible and intuitive organizational metaphor.

The LifeStreams[21] system (**Figure 2.2.1**) uses a timeline as the major organizational metaphor for managing document spaces to replace conventional files and directories. Stream filters and software agents are used to organize, locate, summarize, and monitor incoming information. Lifestream is motivated by the problems of common desktop systems, a standard single-inheritance file hierarchy, difficulty in making use of archived information, difficulty in getting a big picture view, and difficulty in managing schedules and reminders.



Tue Sep 3 10:17:30 1996

David K comments

To: davidg nick
From: efreeman

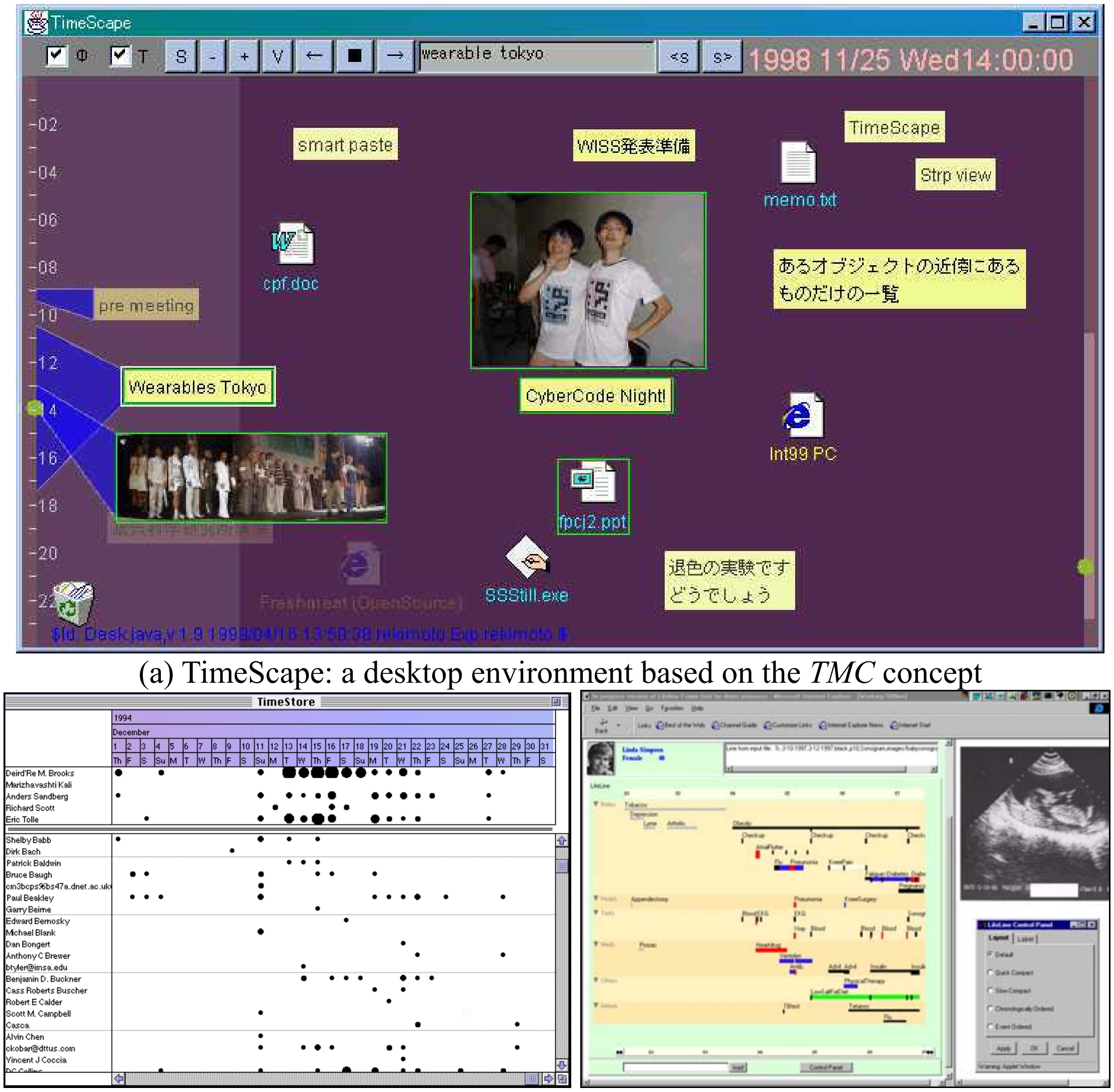
Here are a few comments on the dissertation from David K. I'm not sure what he is referring to in pt 3, I've asked him to clarify.

Eric
spell

Figure 2.2.1 The basic Lifestreams interface: All the documents are arranged based on the timeline.

TimeScape[52] shows an interesting desktop metaphor called TMC(Time Machine Computing). Users can spatially arrange desktop items that can be removed at any time. TimeScape supports time travel to the past or the future of the desktop (**Figure 2.2.2**(a)). The combination of spatial information arrangement and chronological navigation allows users to organize and archive information without being bothered by document folders or file classification problems. TimeStore [38]

(**Figure 2.2.2**(b)) and LifeLines [48] (**Figure 2.2.2**(c)) are other time-based information indexing and retrieval systems.



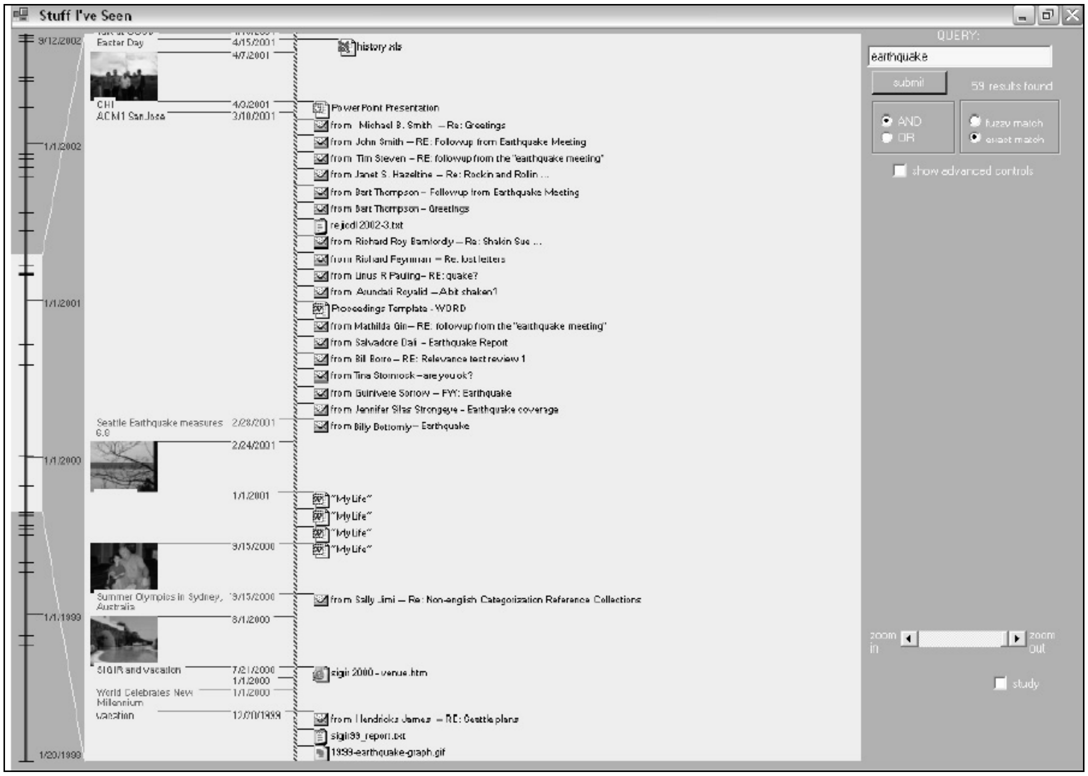
(a) TimeScape: a desktop environment based on the *TMC* concept

(b) TimeStore

(c) Lifeline

Figure 2.2.2 Time-based personal information indexing and retrieval systems

In Stuff I've Seen [70] (SIS) system, Dumais et al. concluded that both the multiplicity of independent applications used to manage information each possessing its own organizational hierarchy (e.g., email, files, web, calendar), and the limited search capabilities in many of them often made it harder to find information on desktop, email store, or intranet than on the web. In addition, they tried to remedy this problem by providing a unified index across different information sources. Furthermore, they explored new ranking and presentation ideas on the assumption that rich contextual cues such as time, author, thumbnails, and previews can be useful because the information is personal and has been seen before. Later, they enhanced SIS interface (**Figure 2.2.3**) to represent the results of searches with an overview-plus-detail timeline visualization and extended a basic time view by adding public landmarks (holidays and important news events) and personal landmarks (photos and important calendar events).



QUERY: earthquake

59 results found

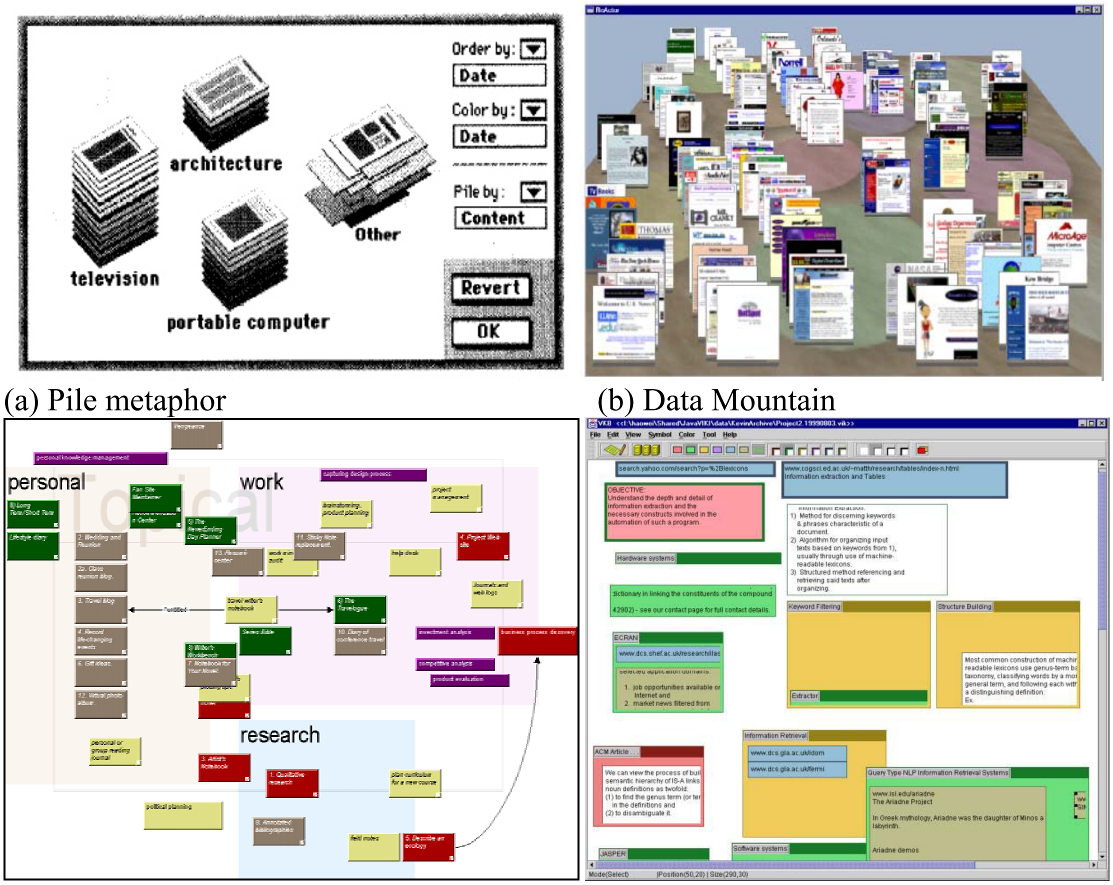
● AND ○ OR

○ fuzzy match ● exact match

history.xls
PowerPoint Presentation
from Michael B. Smith – Re: Greetings
from John Smith – RE: Followup from Earthquake Meeting
from Tim Steven – RE: followup from the "earthquake meeting"
from Janet S. Hazeltine – Re: Rockin and Rollin ...
from Bart Thompson – Followup from Earthquake Meeting
from Bart Thompson – Greetings
rejcdl 2002-3.txt
from Richard Roy Bamlordy – Re: Shakin Sue ...
from Richard Feynman – Re: lost letters
from Linus R Pauling – RE: quake?
from Arundati Royalid – A bit shaken?
Proceedings Template - WORD
from Mathilda Gin – RE: followup from the "earthquake meeting"
from Salvadore Dali – Earthquake Report
from Bill Borro – RE: Relevance test review 1
from Tina Stormock – are you ok?
from Guinivere Sorrow – FYI: Earthquake
from Jennifer Silas Strongeye – Earthquake coverage
from Billy Bottomly – Earthquake
"My Life"
"My Life"
"My Life"
"My Life"
from Sally Jimi – Re: Non-english Categorization Reference Collections
sigir 2000 - venue.htm
from Hendricks James – RE: Seattle plans
sigir99_report.txt
1999-earthquake-graph.gif

Figure 2.2.3 SIS timeline visualization. The overview area at the left shows a timeline with hash marks representing the distribution of the search results over time. The highlighted region of the overview timeline corresponds to the segment of time displayed in the detailed view. To the left of the detailed timeline backbone, beyond basic dates, context is provided with landmarks drawn from news headlines, holidays, calendar appointments, and digital photographs. To the right of the backbone, details of individual search results (represented by icons and titles) are presented.

On the other hand, there have been several research attempts to use different organizing metaphors than a timeline. Richard et al.[39] designed an interesting desktop user interface metaphor, “Pile”, to support the casual organization of documents (**Figure 2.2.4**(a)). The interface design based on the pile concept suggested uses of content awareness for describing, organizing, and filing textual documents. Data Mountain [54] allows users to place documents at arbitrary positions on an inclined plane in a 2.5D desktop virtual environment using a simple 2D interaction technique (**Figure 2.2.4**(b)). This user interface was designed specifically to take advantage of human spatial memory in retrieving a document from a virtual desktop working space.



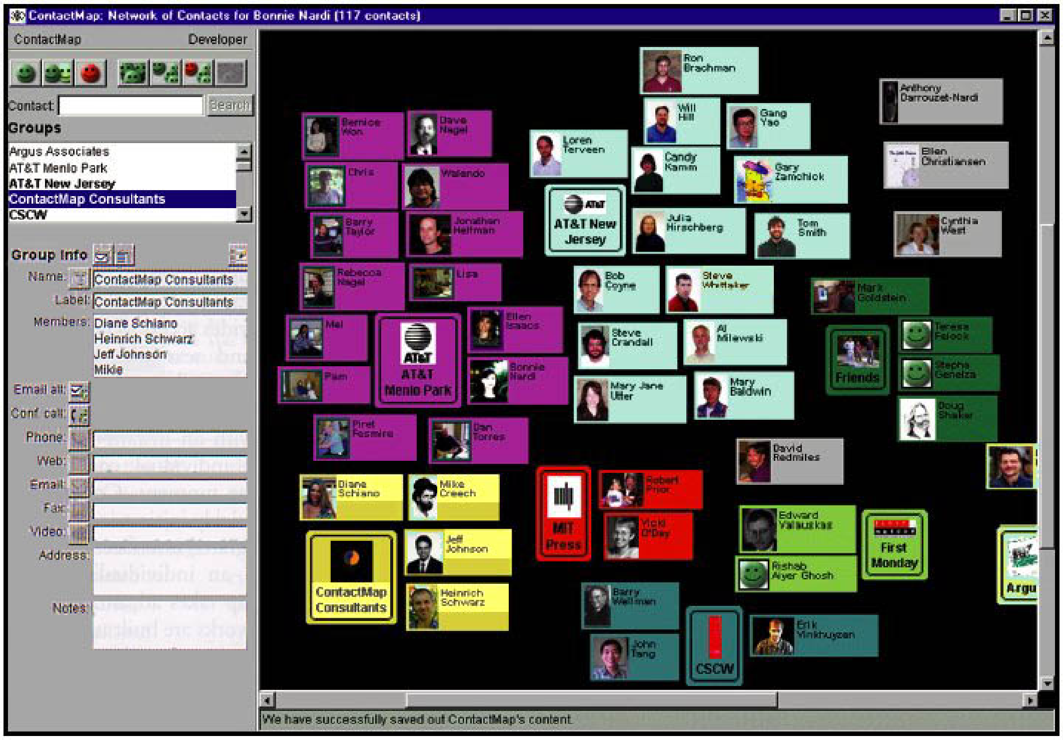
(a) Pile metaphor (b) Data Mountain

(c) TinderBox (d) Visual Knowledge Builder

Figure 2.2.4 Visual information work spaces using spatial organization

Tinderbox[69] (**Figure 2.2.4**(c)), Niagara[78], Visual Knowledge Builder[58] (**Figure 2.2.4**(d)) are visual information workspaces (either personal or collaborative) supporting the collection, organization, and annotation of information. These tools make use of spatial cues as well as patterns in visual space to represent the characteristics of and interrelationships within the information.

On the other hand, ContactMap[68] uses the models of personal social networks to provide an organizing principle for advanced user interfaces that afford information and communication services in a single integrated system (**Figure 2.2.5**). This research is based on the fact that people invest considerable effort in maintaining links with networks of colleagues, acquaintances and friends, and that these networks are a significant organizing principle for work and information.



ContactMap: Network of Contacts for Bonnie Nardi (117 contacts)

Contact:

Groups

Argus Associates
AT&T Menlo Park
AT&T New Jersey
ContactMap Consultants
CSCW

Group Info

Name: ContactMap Consultants

Label: ContactMap Consultants

Members: Diane Schiano
Heinrich Schwarz
Jeff Johnson
Mikie

We have successfully saved out ContactMap's content.

Figure 2.2.5 ContactMap allows users to arrange their individual social networks in a visual map of individual contacts and groups: each node affords communication functions enabling users to retrieve current and archived information associated with them.

A few research efforts have been dedicated to index personal media with high-level attributes (either manually defined or automatically generated) and make use of them to construct flexible hierarchies or multiple categorizations which provide users various viewpoints of personal information organizations.

Presto [20] is a prototype document management system providing rich interaction with documents through user-level document attributes (**Figure 2.2.6**). Document attributes capture the multiple roles that a single document might play, and allow users to reorganize their document space for the task at hand. The model focuses on the fact that users interact with their documents in terms of high-level attributes that are provided and fluidly managed by the users themselves.

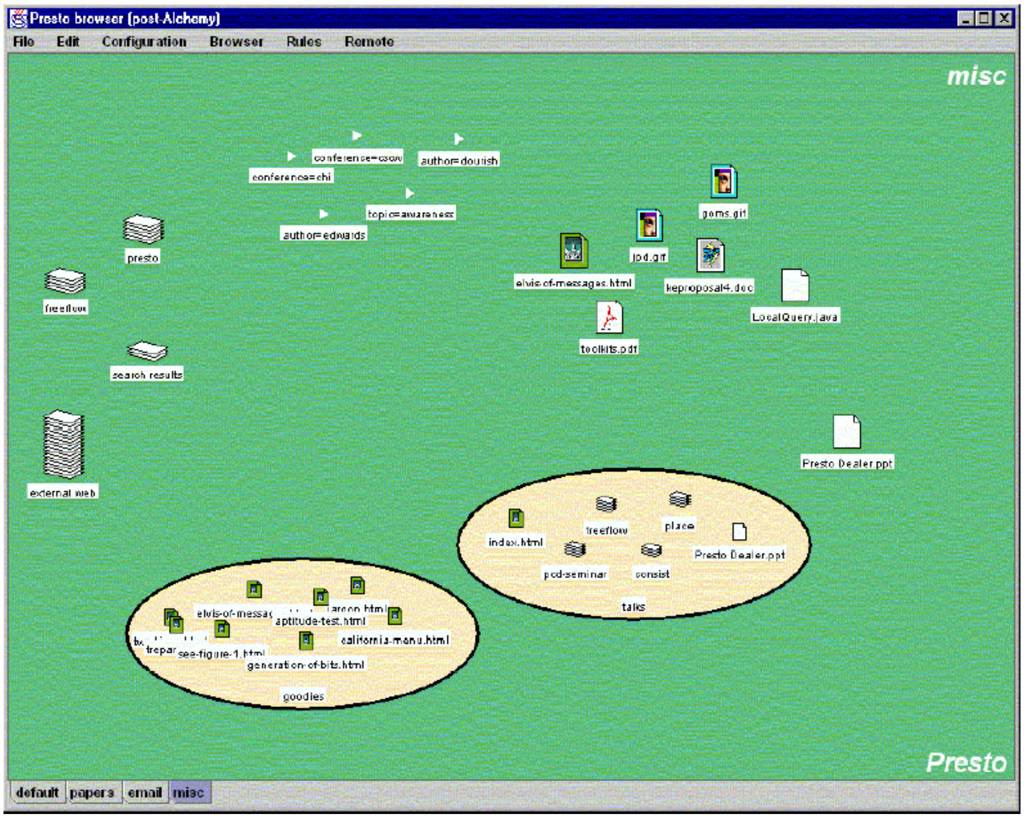
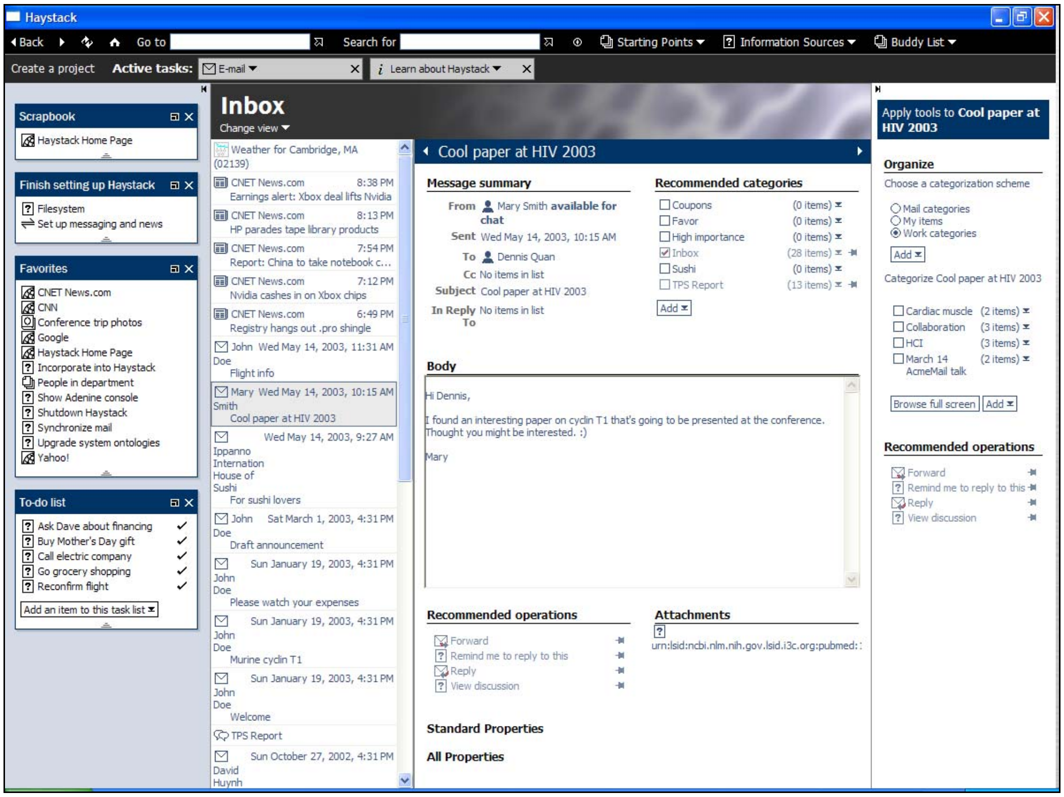


Figure 2.2.6 Presto Browser: In addition to a number of documents, the desktop contains four closed collections, two opened collections, and a set of predefined attributes.

The Semantic File System[24] is an information storage system that provides flexible associative access to the system's contents by automatically extracting attributes from files with file type specific transducers. Associative access is provided by a conservative extension to existing tree-structured file system protocols and by protocols that are designed specifically for content-based access. Compatibility with existing file system protocols is provided by introducing the concept of a "virtual directory". In this system, virtual directory names are interpreted as queries and thus provide flexible associative access to files and directories in a manner compatible with existing file systems.



To-do list:

- Ask Dave about financing ✓
- Buy Mother's Day gift ✓
- Call electric company ✓
- Go grocery shopping ✓
- Reconfirm flight ✓

Add an item to this task list

Inbox:

Weather for Cambridge, MA (02139)

CNET News.com 8:38 PM
Earnings alert: Xbox deal lifts Nvidia

CNET News.com 8:13 PM
HP parades tape library products

CNET News.com 7:54 PM
Report: China to take notebook c...

CNET News.com 7:12 PM
Nvidia cashes in on Xbox chips

CNET News.com 6:49 PM
Registry hangs out .pro shingle

John Doe Wed May 14, 2003, 11:31 AM
Flight info

Mary Smith Wed May 14, 2003, 10:15 AM
Cool paper at HIV 2003

Ippanno Internation House of Sushi Wed May 14, 2003, 9:27 AM
For sushi lovers

John Doe Sat March 1, 2003, 4:31 PM
Draft announcement

John Doe Sun January 19, 2003, 4:31 PM
Please watch your expenses

John Doe Sun January 19, 2003, 4:31 PM
Murine cyclin T1

John Doe Sun January 19, 2003, 4:31 PM
Welcome

TPS Report

David Huynh Sun October 27, 2002, 4:31 PM

Message summary

From Mary Smith **available for chat**
Sent Wed May 14, 2003, 10:15 AM
To Dennis Quan
Cc No items in list
Subject Cool paper at HIV 2003
In Reply To No items in list

Recommended categories

□ Coupons	(0 items)
□ Favor	(0 items)
□ High importance	(0 items)
☑ Inbox	(28 items)
□ Sushi	(0 items)
□ TPS Report	(13 items)

Add

Body

Hi Dennis,

I found an interesting paper on cyclin T1 that's going to be presented at the conference. Thought you might be interested. :)

Mary

Recommended operations: Forward, Remind me to reply to this, Reply, View discussion

Attachments: urn:lsid:ncbi.nlm.nih.gov.lsid.i3c.org:pubmed:

Choose a categorization scheme: ○ Mail categories ○ My items ● Work categories

Categorize Cool paper at HIV 2003

□ Cardiac muscle	(2 items)
□ Collaboration	(3 items)
□ HCI	(3 items)
□ March 14 AcmeMail talk	(2 items)

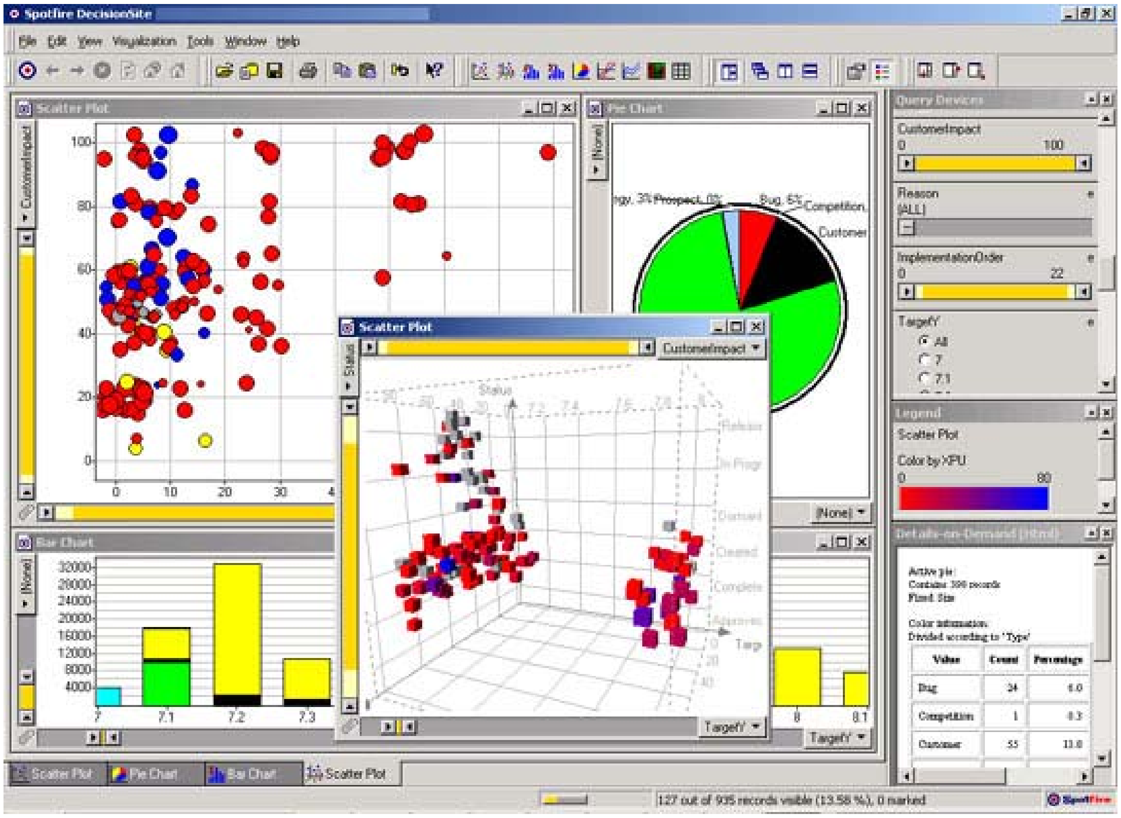
Figure 2.2.7 Haystack is designed to let users manage all of their information in the way that makes the most sense to them. It allows users to define whichever arrangements of, connections between, and views of information they find most effective.

Haystack[1][50] (**Figure 2.2.7**) uses a semistructured data model to describe the connection between different documents in a user's corpus as well as the metadata concerning each document. It adopts a category-based organization and navigation scheme that allows information to be placed in multiple thematic "bins", or categories, simultaneously. This research is based on the belief that information inherently has multiple, relevant categories that users can readily identify and that allowing multiple categories lets users organize documents in a more intuitive, richer information space.

2.3 Flexible and Generalized Information Visualization

Users often need various views in order to handle the different types of information, different visualizations, and multiple perspectives for different aspects of an information task [13][44]. However, it is difficult for users to find generalized visualization tools satisfying their demands, or to change visualizations dynamically according to their mental models or specific tasks.

SpotFire [65] is one of the most successful generalized information visualization and visual data-mining tools. Easy import/export of data, rapid change to axes, color coding or size coding, and direct data manipulation [3][2][75] have made it a leading generalized data visualization tool. However, it has a limitation in that all data must be projected onto the mathematical 1D, 2D or 3D spaces as bar charts, pie charts, or scattergrams (**Figure 2.3.1**) even though they can be visualized more intuitively in other simple user-friendly display representations such as map, calendar, genealogy trees, etc. Like other successful statistical data mining tools such as SAS JMP and SPSS Diamond [**60**], Spotfire is more appropriate for visualizing scientific and quantitative data rather than metadata-based (nominal attributes based) personal media data.



Active pie:
Contains 398 records
Fixed Size

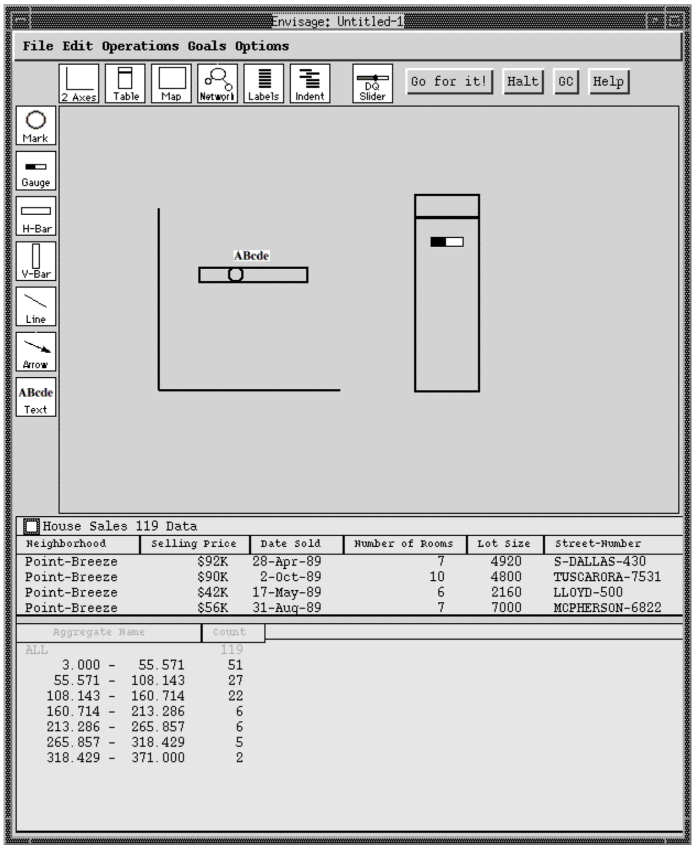
Color information:
Divided according to 'Type'

Value	Count	Percentage
Bug	24	6.0
Competition	1	0.3
Customer	55	13.8

127 out of 935 records visible (13.58 %), 0 marked

Figure 2.3.1 SpotFire for scientific and statistical data visualization

Visage [56] is a prototype user interface environment that enables users to move and combine interesting subsets of information across the isolating boundaries imposed by different applications. Visage and SageBrush [56][57] enables users to assemble combinations of graphical elements (e.g. axes, lines, text, etc.) and map these elements to the data that users wish to visualize. With Visage's "information-centric" approach, users can drag-and-drop data items between visualizations to display them in different ways. The Visage VQE component also coordinates dynamic queries across all visualizations within the VQE window. The Visage SAGE component generates different types of visualizations. Users specify the visualization by associating data attributes with a visual element.



□ House Sales 119 Data

Neighborhood	Selling Price	Date Sold	Number of Rooms	Lot Size	Street-Number
Point-Breeze	$92K	28-Apr-89	7	4920	S-DALLAS-430
Point-Breeze	$90K	2-Oct-89	10	4800	TUSCARORA-7531
Point-Breeze	$42K	17-May-89	6	2160	LLOYD-500
Point-Breeze	$56K	31-Aug-89	7	7000	MCPHERSON-6822

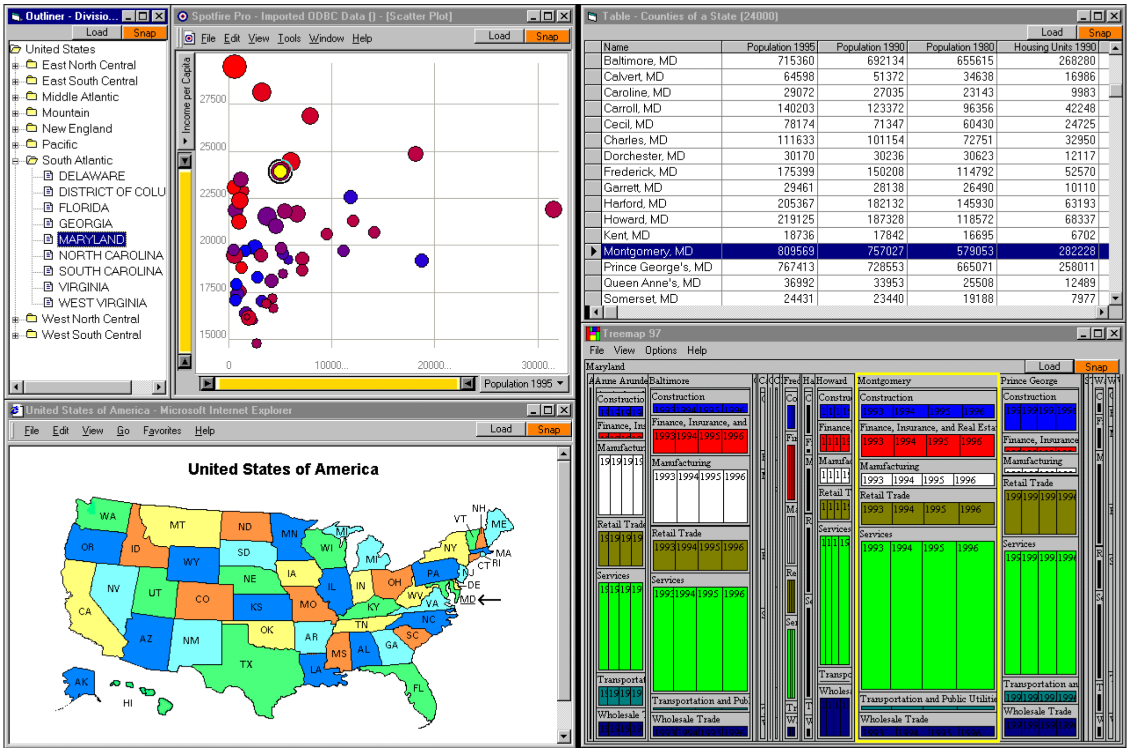
Aggregate Name	Count
ALL	119
3.000 - 55.571	51
55.571 - 108.143	27
108.143 - 160.714	22
160.714 - 213.286	6
213.286 - 265.857	6
265.857 - 318.429	5
318.429 - 371.000	2

Figure 2.3.2 SageBrush. Brush's interface consists of a design work area (center) into which users drag prototypes (top), graphemes (left), and data names (bottom).

Snap-together [44] visualization enables users to rapidly and dynamically construct coordinated-visualization interfaces, customized for their data, without programming. Users load data into desired visualizations, then create coordination between them, such as brushing and linking, overview and detail, and drill down **Figure 2.3.3**).

On the other hand, there was an early paradigm of separating data and visualization to enable more flexible information visualization. The Model/View/Controller (MVC) framework [34] was first introduced in Smalltalk-80.

In the MVC paradigm (**Figure 2.3.4**), the user input, the modeling of the external world, and the visual feedback to the user are explicitly separated and handled by three types of object, each specialized for its task. The "view" manages the graphical and/or textual output to the portion of the bitmapped display that is allocated to its application. The "controller" interprets the mouse and keyboard inputs from the user, commanding the model and/or the view to change as is appropriate. Finally, the "model" manages the behavior and data of the application domain, responds to requests for information about its state, and responds to instructions to change its state. The formal separation of these three tasks makes GUI (Graphical User Interface) design and implementation more flexible and powerful and also shows the possibility of applying various views to the same model according to the users' mental models and personal ontologies.



Name	Population 1995	Population 1990	Population 1980	Housing Units 1990
Baltimore, MD	715360	692134	655615	268280
Calvert, MD	64598	51372	34638	16986
Caroline, MD	29072	27035	23143	9983
Carroll, MD	140203	123372	96356	42248
Cecil, MD	78174	71347	60430	24725
Charles, MD	111633	101154	72751	32950
Dorchester, MD	30170	30236	30623	12117
Frederick, MD	175399	150208	114792	52570
Garrett, MD	29461	28138	26490	10110
Harford, MD	205367	182132	145930	63193
Howard, MD	219125	187328	118572	68337
Kent, MD	18736	17842	16695	6702
Montgomery, MD	809569	757027	579053	282228
Prince George's, MD	767413	728553	665071	258011
Queen Anne's, MD	36992	33953	25508	12489
Somerset, MD	24431	23440	19188	7977

Figure 2.3.3 Snap-together visualization: A coordinated-visualization interface for browsing census data.

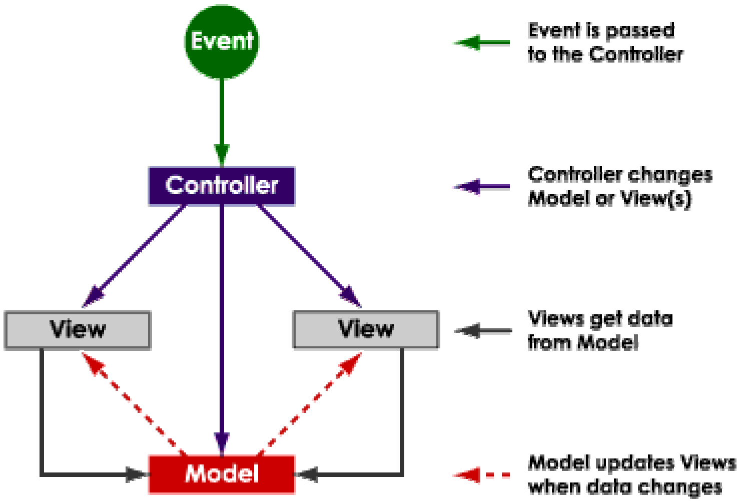

Event
Controller
View
View
Model
Event is passed to the Controller
Controller changes Model or View(s)
Views get data from Model
Model updates Views when data changes

Figure 2.3.4 The MVC abstraction diagram: Events typically cause a controller to change a model, or view, or both. Whenever a controller changes a model's data or properties, all dependent views are automatically updated. Similarly, whenever a controller changes a view, for example, by revealing areas that were previously hidden, the view gets data from the underlying model to refresh itself.

2.4 Summary

Due to extraordinary progress in all of the technologies for *cyberization* and users' eagerness to capture, digitize and keep personal information on their individual machines for personal history or archive, [40] [8] [45] [73] [37] [17] have studied ways to capture personal information automatically in daily life as well as ways to store them in the system for later use. However, there was an insufficient amount of focus placed on the management of the collected information.

On the other hand, there has been a great deal of research which attempts to solve problems related to indexing, retrieval, and management of files and documents by supporting a more flexible and intuitive organizational metaphor. [21] [52] [38] [48] [70] are the systems that were motivated by the problems of a standard single-inheritance file hierarchy. They made use of a timeline as a new organizing metaphor for the document management system and resolved some of the problems which resulted from the rigid folder hierarchy structure. Although their organizing metaphor looks more intuitive and user-oriented, they simply replaced one superordinate aspect of the document (its location in the hierarchy) with another (its location in the timeline). Their organizing metaphor is still fixed and rigid and cannot be applied to the problems that require multiple perspectives of the data.

[54] [69] [78] [58] [68] make use of spatial cues and patterns in visual space to represent the characteristics of and interrelationships within the information. These tools also take advantage of human spatial memory in retrieving documents from a virtual desktop working space. Although spatial organization of information enables users to retrieve documents quickly, it is not easy for users to represent multiple organizations based on the specific tasks or their various mental models. Furthermore, there is no good way to represent an organizational hierarchy with these tools.

Unlike other systems, [20] [24] [1] [50] tried to index documents with high-level attributes (either user-defined or automatically generated) and make use of them in constructing flexible hierarchy or multiple categorization. This approach enables flexible organization of information and enriches the representations of information due to the high level attributes. The appropriate visualization as well as graphical user interface for these systems can leverage the power of document management.

[65] [56] [57] [44] [34] are the tools and a framework to support flexible and generalized information visualization of data. Although they provide multiple perspectives for different aspects of an information task, most of them are designed to deal with the scientific and statistical data stored in the relational database table rather than the user-defined high level attributes of personal media data.

Chapter 3:

Preliminary Works: Personal Media Management Tools

The author of this dissertation has been actively engaged in research dealing with personal media such as PhotoFinder (a tool for organizing, annotating, and exploring the personal photo library), PhotoFinder Kiosk (an extension of PhotoFinder with network support and group annotation), and Multimedia Bulletin Board (a prototype of the next generation asynchronous information system for rich communication and collaboration among people using multimedia data). The facts found and the lessons learned from these research formed a basis for the concept of *Semantic Regions* and led to the development of a personal media management tool, MediaFinder.

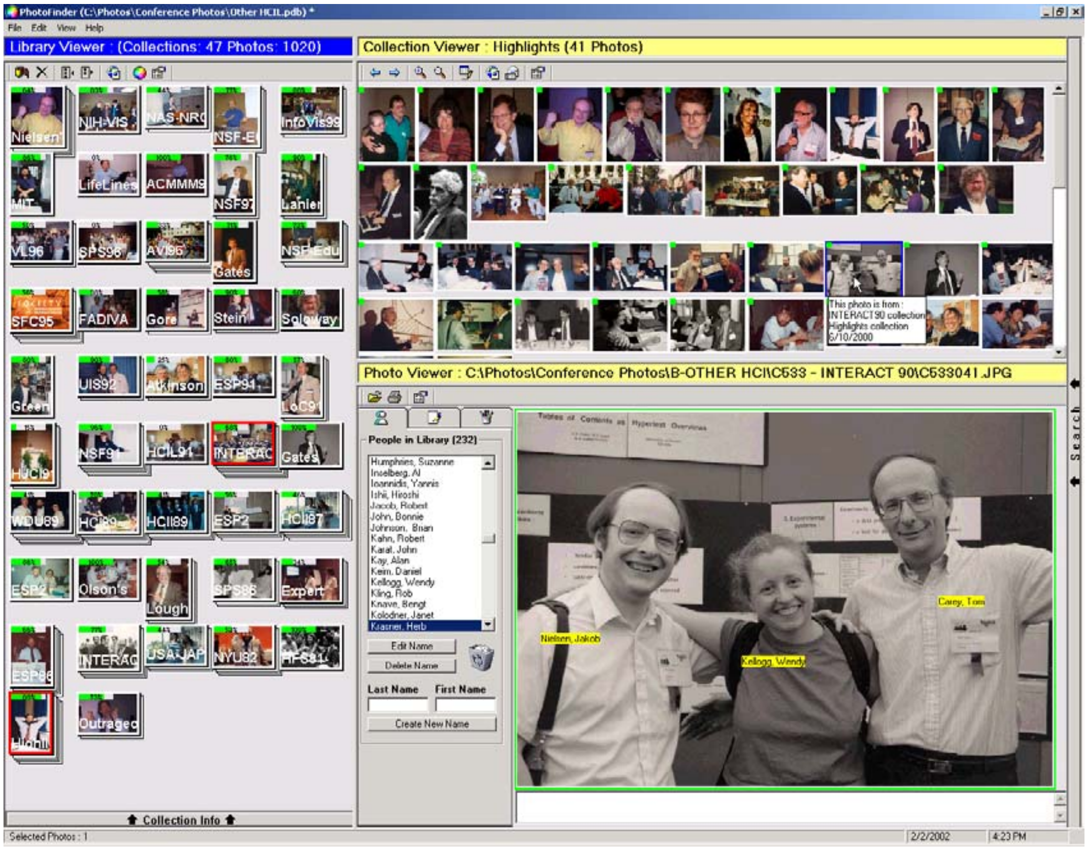
3.1 PhotoFinder3

The author has been developing the PhotoFinder [33] prototypes (http://www.cs.umd.edu/hcil/photolib) during the past four years, as part of a research effort on Personal Photo Libraries. The goal of the PhotoFinder project was to develop an understanding of user needs, appropriate tasks, and innovative designs for consumer users of digital photos. As digital cameras, scans of existing photos, PhotoCDs, and photos by email become increasingly common, users may have to manage hundreds and possibly even thousands of photos. The goals of users who actively deal with digital photos are to view, explore, locate, reorganize, and then use photos of interest. The PhotoFinder project led to the development of a conceptual model of a library having a set of collections, with each collection having a set of photos. Photos can be included in multiple collections. Collections and individual photos can be annotated with free text fields plus date and location fields stored in a database. The PhotoFinder interface has three main windows (**Figure 3.1.1**):

- *Library viewer:* Shows a representative photo for each collection, with a stack representing the number of photos in each collection.
- *Collection viewer:* Shows thumbnails of all photos in the collection. Users can move the photos around, enlarge them all or individually, cluster them, or present them in a compact manner. Users can see thumbnails in several different types of display modes (sequential display, histogram display, scatterplot display, user-defined display, spiral display, 3D timeline display, 3D merry-go-round display, etc.).
- *Photo viewer:* Shows an individual photo in a resizable window. A group of photos can be selected in the Collection viewer and dragged to the Photo viewer to produce an animated slide show.

In addition to the organization of photos, a strong emphasis was put on recording and searching by metadata in each photo. Since PhotoFinder was designed for personal use, name of people in each photo was focused initially. It is believed that a personal photo library might contain repeated images of the same people at different events, and estimated 100-200 identifiable people in 10,000 photos. The many-to-many relationship between photos and people is mediated by the Appearance relation (**Figure 3.1.2**) that stores the identification of all the people who appear in each photo.

Such a database would support accurate storage of information, but it was recognized that the tedious data entry problem would prevent most users from typing in names for each photo. Furthermore, the inconsistency in names is quickly a problem with misspellings or variant names (for example, Bill, Billy, William) undermining the success of search.



PhotoFinder (C:\Photos\Conference Photos\Other HCIL.pdb) *

Library Viewer : (Collections: 47 Photos: 1020)

Collection Viewer : Highlights (41 Photos)

This photo is from :
INTERACT90 collection
Highlights collection
6/10/2000

Photo Viewer : C:\Photos\Conference Photos\B-OTHER HCI\C533 - INTERACT 90\C533041.JPG

People in Library (232)

Humphries, Suzanne
Inselberg, Al
Ioannidis, Yannis
Ishii, Hiroshi
Jacob, Robert
John, Bonnie
Johnson, Brian
Kahn, Robert
Karat, John
Kay, Alan
Keim, Daniel
Kellogg, Wendy
Kling, Rob
Knave, Bengt
Kolodner, Janet
Krasner, Herb

Last Name
First Name

Selected Photos : 1

2/2/2002
4:23 PM

Figure 3.1.1 PhotoFinder3 display with Library Viewer on the left, Collection Viewer with thumbnails on the upper right, and Photo Viewer on the lower right. Library Viewer shows a representative photo for each collection with collection information such as the collection name, description, date range, location, number of photos, and percentage of annotations and captions. Collection Viewer shows the photo thumbnails of the search result or the selected collection. Photo Viewer shows the real photo image of the selected thumbnail with annotations, captions, and other individual photo information. The list of names, showing all of the annotated people in the library, is used for search and direct annotation.

A second challenge we faced was that the list of names of people appearing in a photo could often be difficult to associate with individuals, especially in group shots. Textual captions often indicate left-to-right ordering in front and back rows, or give even more specific identification of who is where.

To cope with these challenges, we developed the concept of direct annotation: selectable, dragable labels that can be placed directly on the photo. Users can select from a scrolling or pop-up list and drag by mouse or touch screen. This applies direct manipulation principles [59] that avoid the use of a keyboard, except to enter a name the first time it appears. The name labels can be moved or hidden, and their presence is recorded in the database in the Appearance relation with an X-Y location, based on an origin in the upper left hand corner of the photo.

This simple rapid process also allows users to annotate at will. They can add annotations when they first see their photos on the screen, when they review them and make selections, or when they are showing them to others. This easy design and continuous annotation facility encourages users to do more annotation.

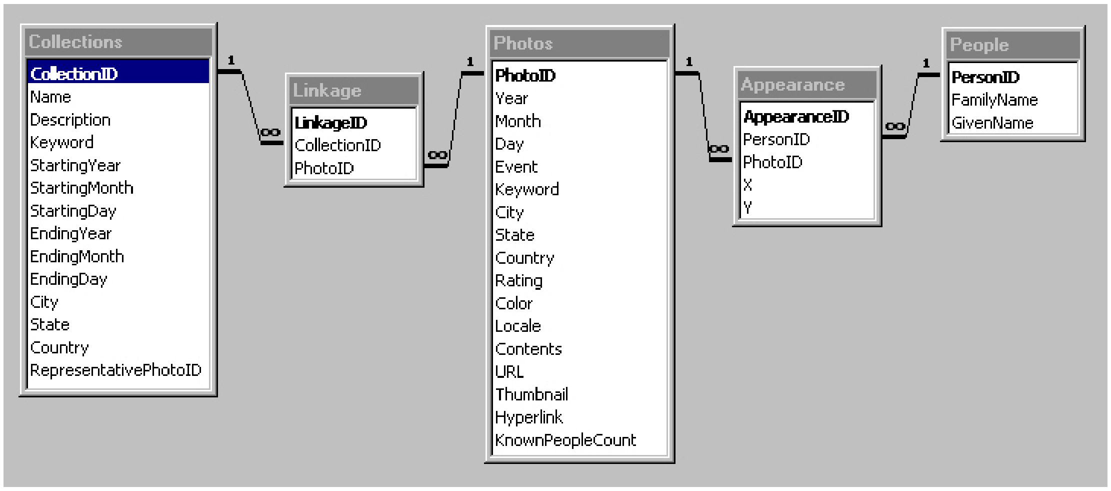
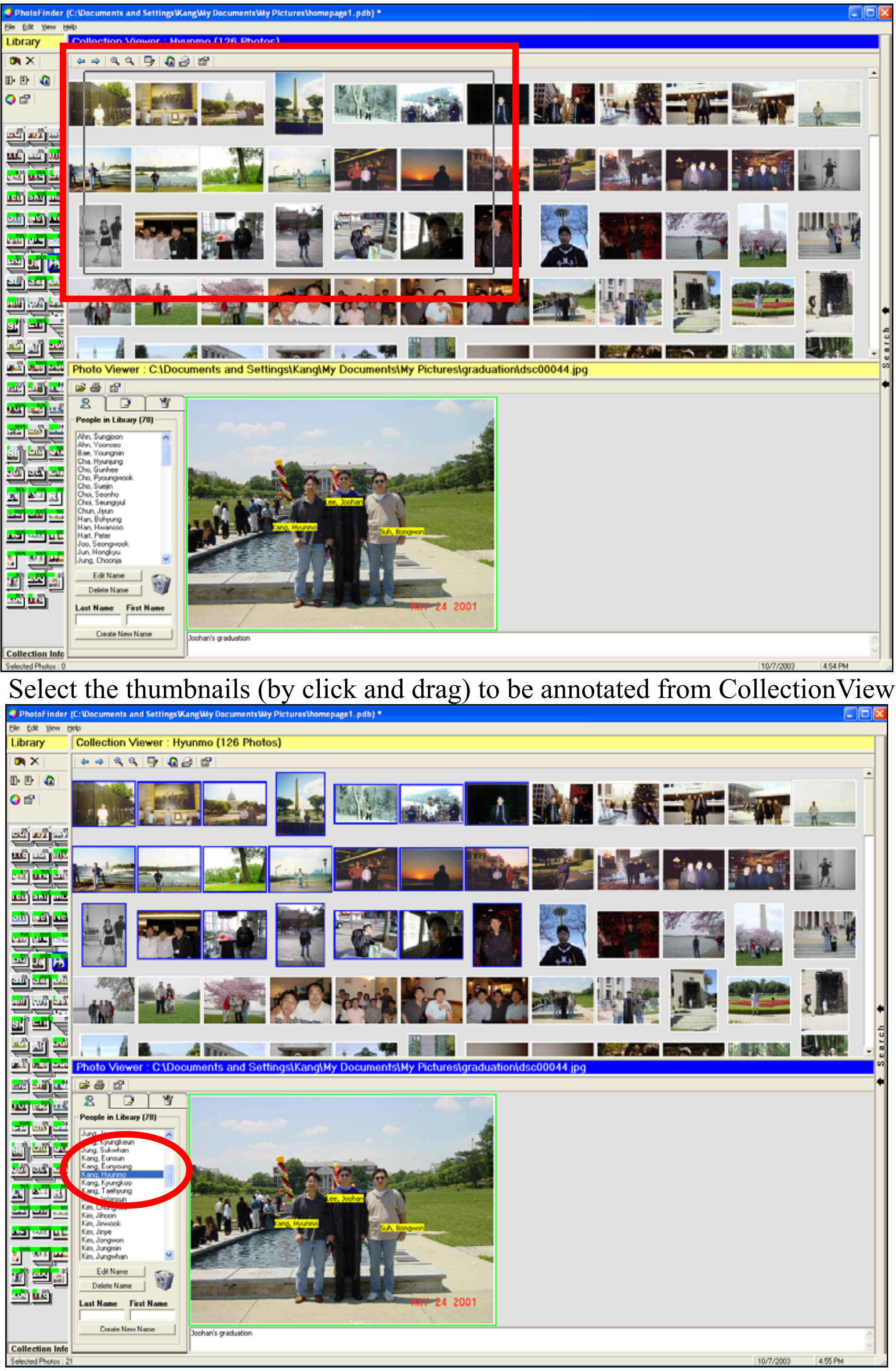


Figure 3.1.2 The schema of Photo Library database

Further reinforcement for annotation is given by showing a small green tag on the top left corner of photos in the Collection viewer. Users will then be able to see how much they have accomplished and which photos are still in need of annotation. Other features include the capacity to resize the labels, change fonts, and change colors. A valuable accelerator is bulk annotation [35], in which a group of photos is selected and then the same label is applied to every photo with one action, although individual placement might still be needed (**Figure 3.1.3**).

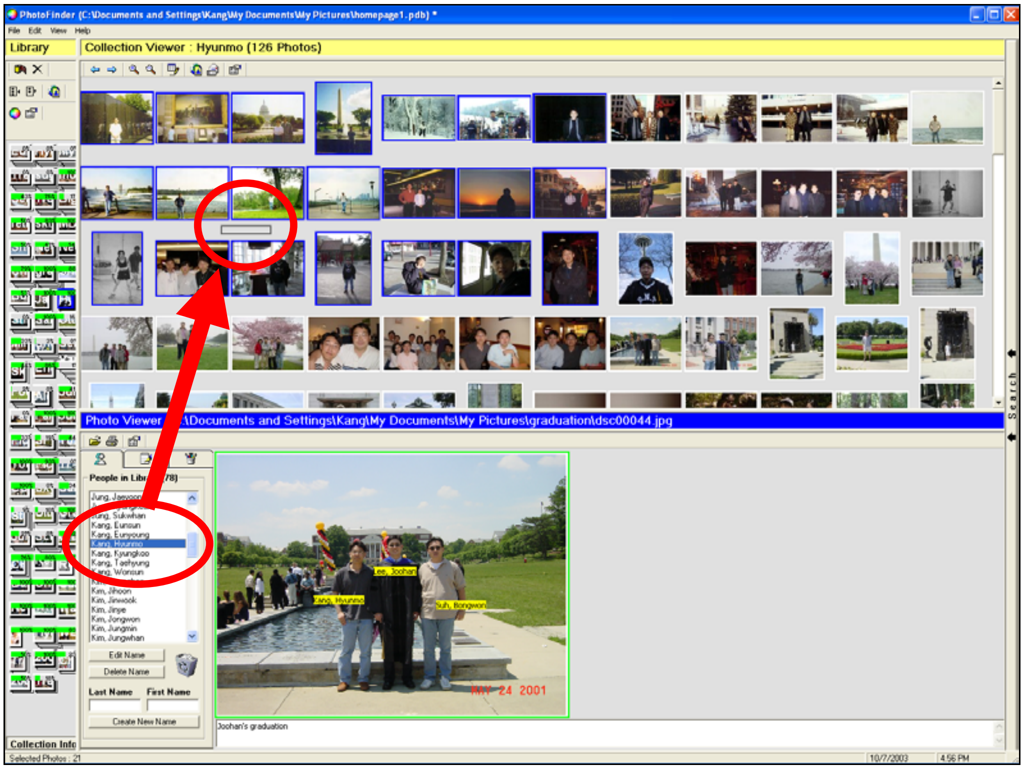
Direct annotation has been extended in the PhotoFinder 4 so that drag and drop annotation for any kind of object in a photo (car, house, bicycle), map (cities, states, lakes), or painting (brushstroke, signature, feature) can be possible. Annotation about the overall image, such as type of photo (portrait, group, landscape), map (highway, topographic, urban), or painting (impressionist, abstract, portrait) is possible by allowing users to define their own categories and values (**Figure 3.1.4**).



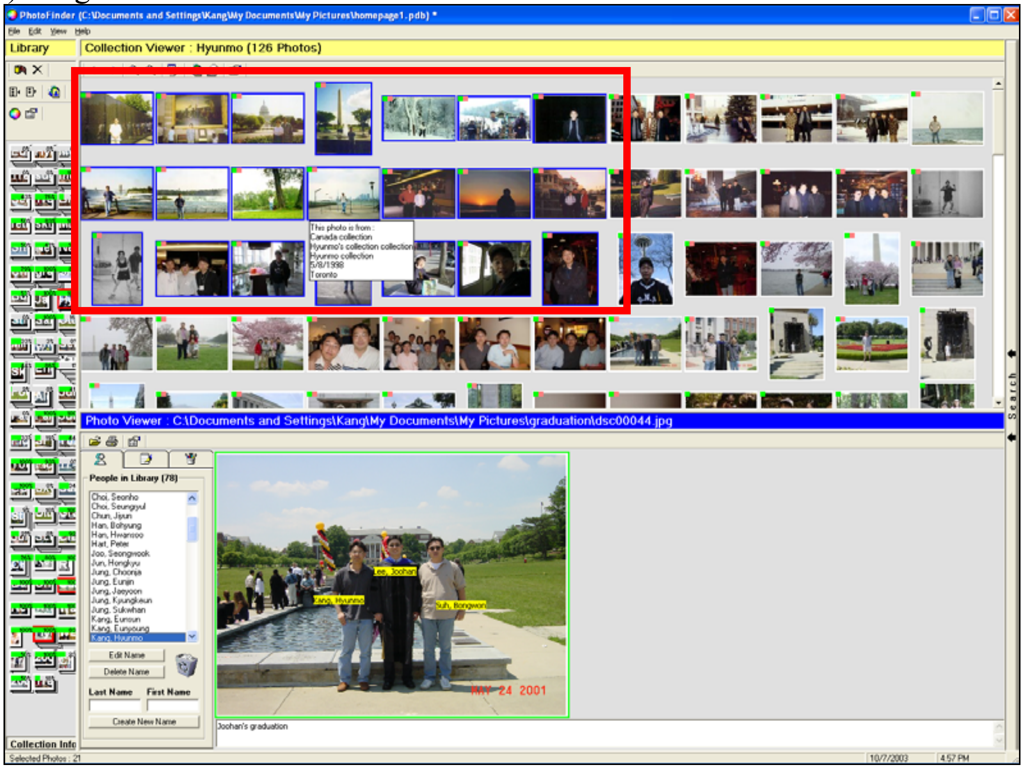
(a) Select the thumbnails (by click and drag) to be annotated from CollectionViewer

(b) Select a name form the list located at the PhotoViewer

Figure 3.1.3 Bulk annotation: multiple photos can be annotated with one drag and drop action just as direct annotation (Continued)



(c) Drag the name label onto the selected thumbnails



(d) Bulk annotation completed

Figure 3.1.3 Bulk annotation: multiple photos can be annotated with one drag and drop action just as direct annotation

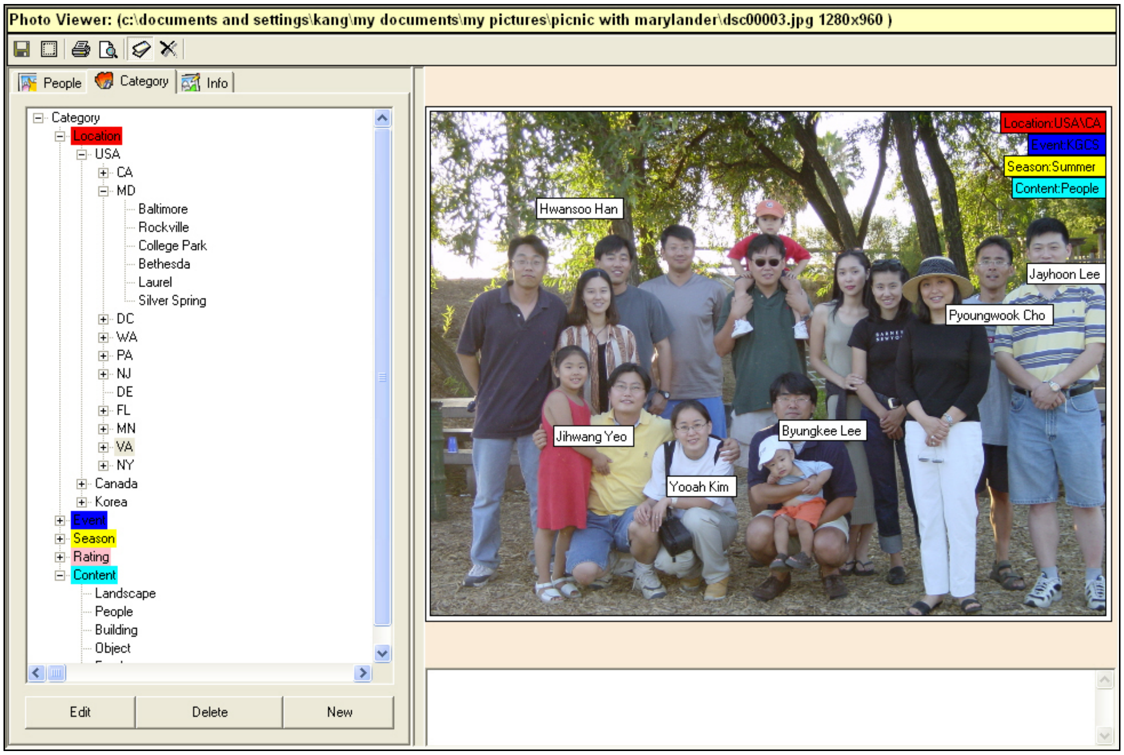
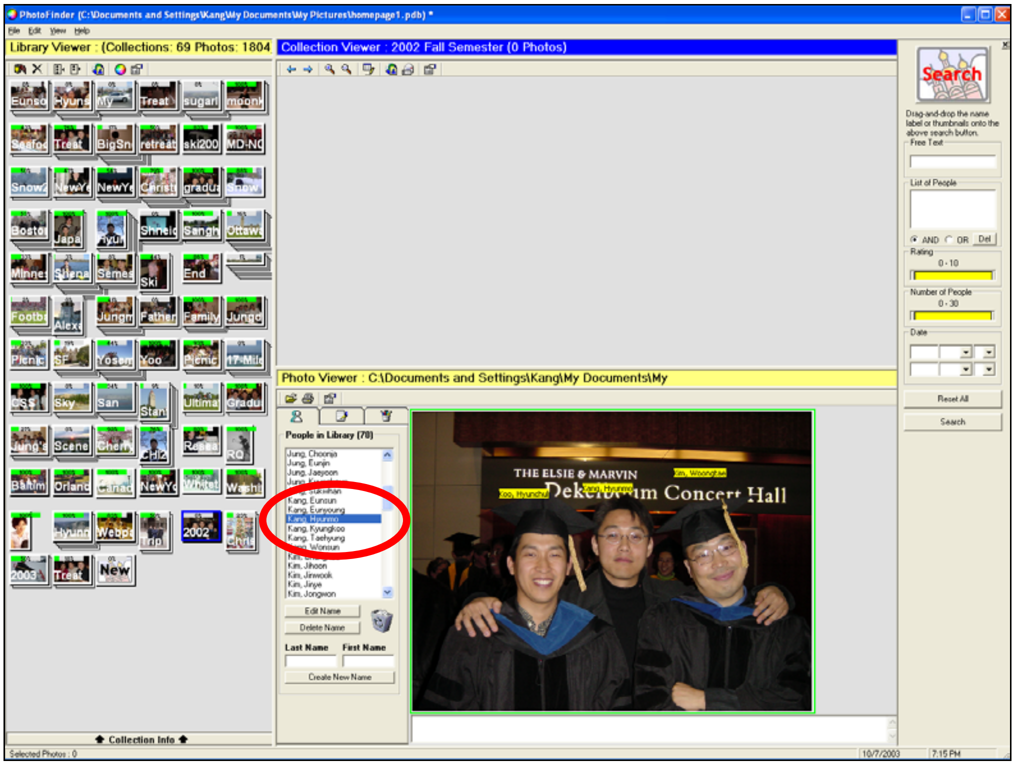
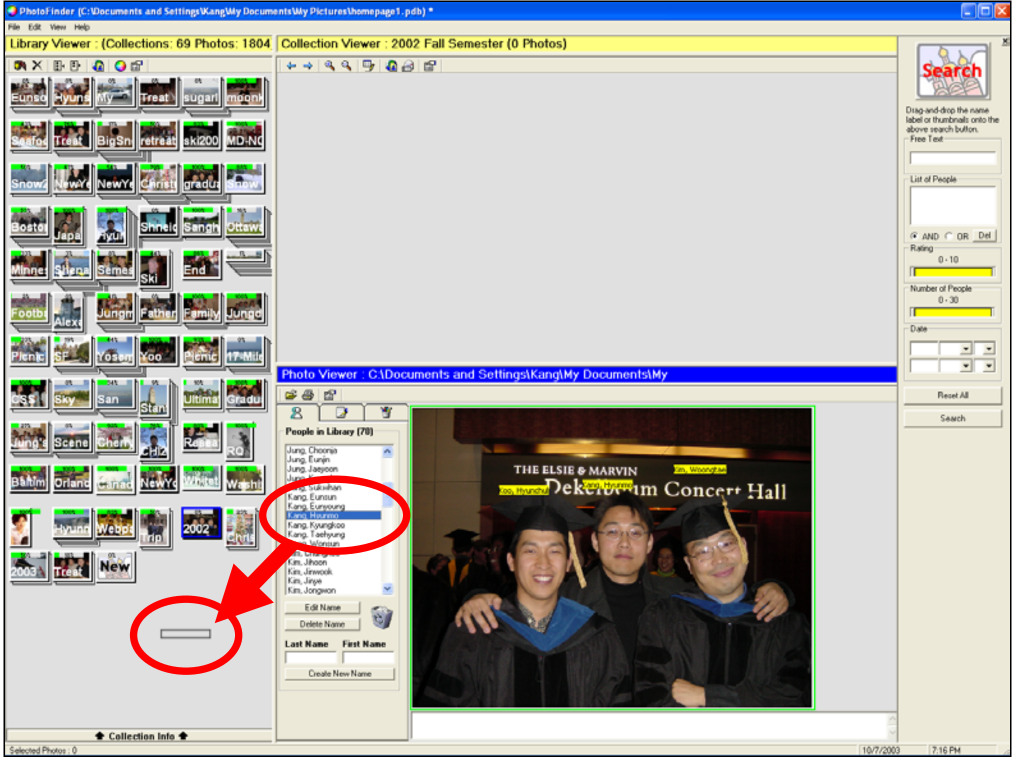


Figure 3.1.4 PhotoFinder 4 extends direct annotation with user-defined category labels. In the left control panel, users can create their own category tags and use them to annotate photos.

Searching and browsing become more effective once annotations are included in the photo database. The obvious task is to see all photos that include an individual. This has been implemented by simply dragging the name from the list into the Library Viewer or to a designated label area. The PhotoFinder finds and displays all photos in which that name appears in a label (**Figure 3.1.5**). Drag and drop search can also be used for filtering out the search result by dragging a label onto the Collection viewer. More complex searches are possible using the search control panel located at the right side of PhotoFinder. All of the specified attribute values in the search control panel are joined conjunctively. PhotoFinder3 does not support disjunctive or complement queries except the people attribute.

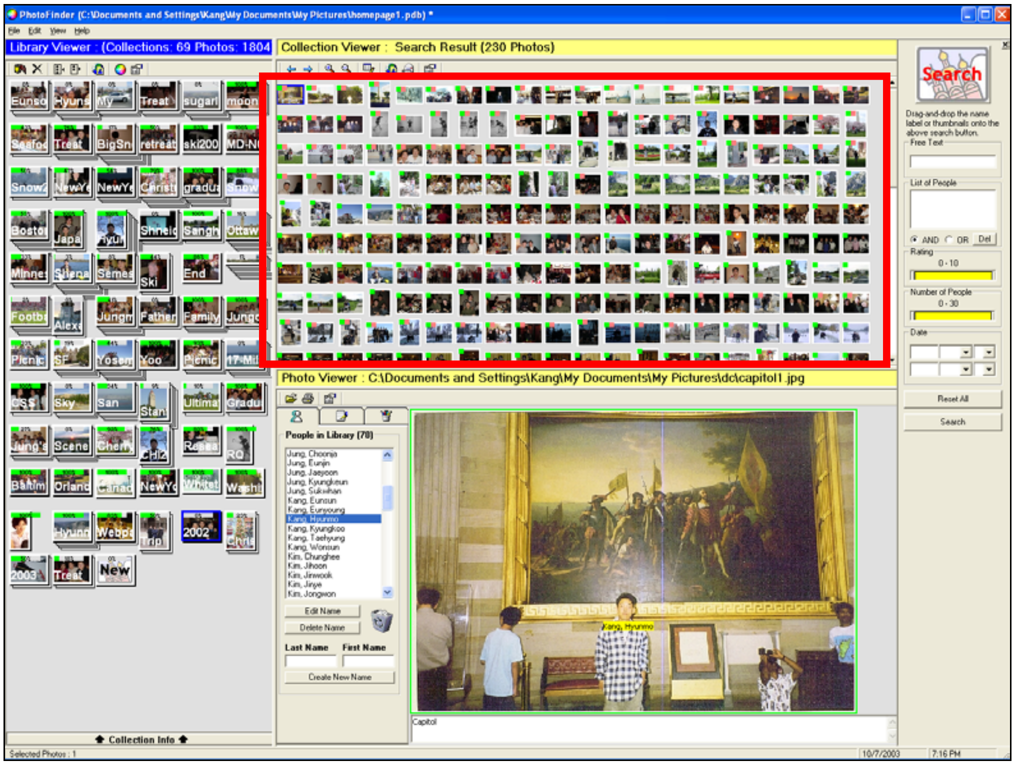


(a) Select a name from the list located at the PhotoViewer

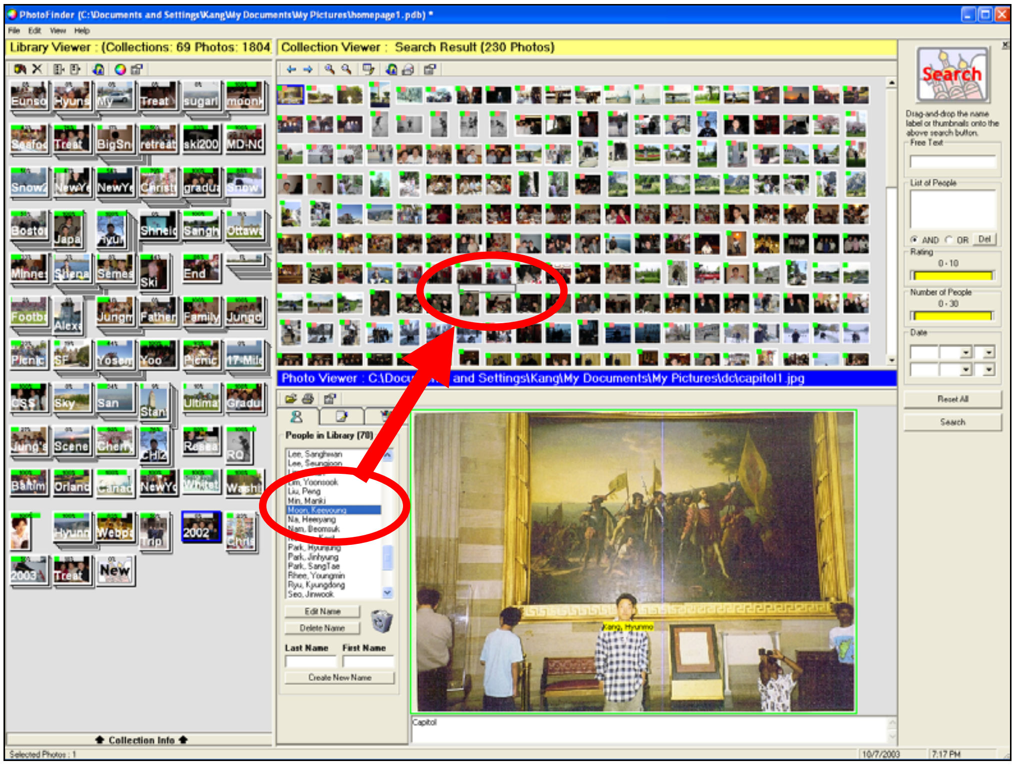

PhotoFinder (C:\Documents and Settings\Kang\My Documents\My Pictures\homepage1.pdb) *
File Edit View Help
Library Viewer : (Collections: 69 Photos: 1804
Collection Viewer : 2002 Fall Semester (0 Photos)
Photo Viewer : C:\Documents and Settings\Kang\My Documents\My
People in Library (78)
Kang, Hyunmo
Edit Name
Delete Name
Last Name
First Name
Create New Name
Search
Reset All
Search
Collection Info
Selected Photos : 0
10/7/2003
7:16 PM

(b) Drag the name onto the LibraryViewer

Figure 3.1.5 Searching the annotated photos in PhotoFinder (Continued)

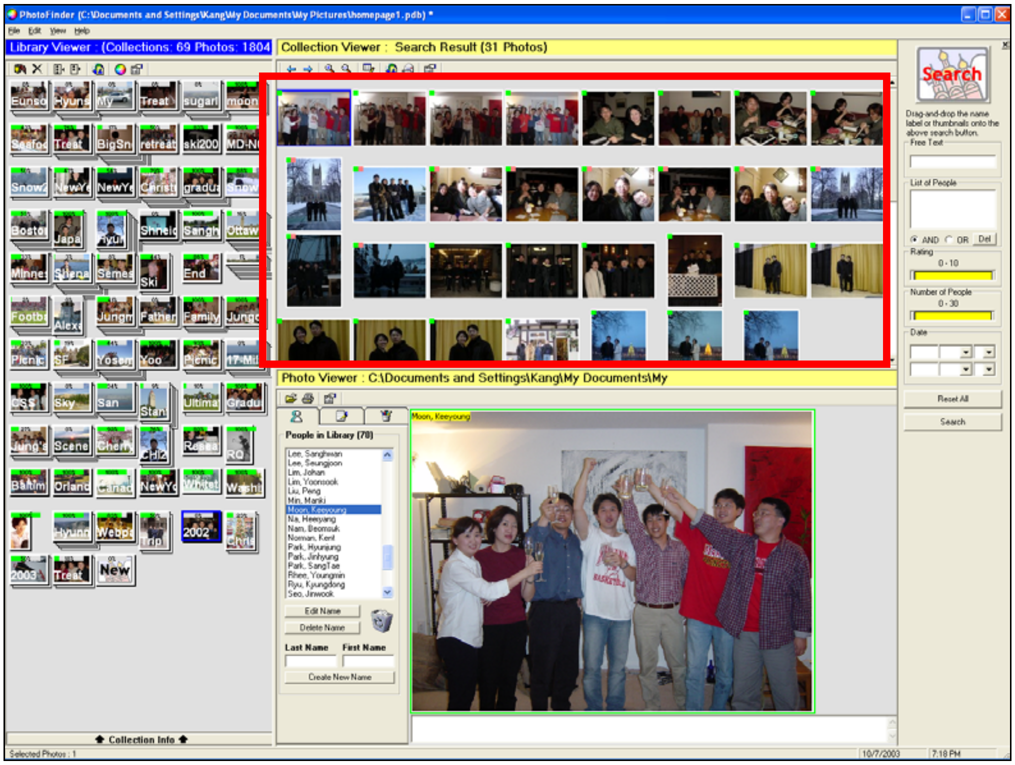


(c) The CollectionViewer shows the search results and the PhotoViewer shows the first photo of the search results

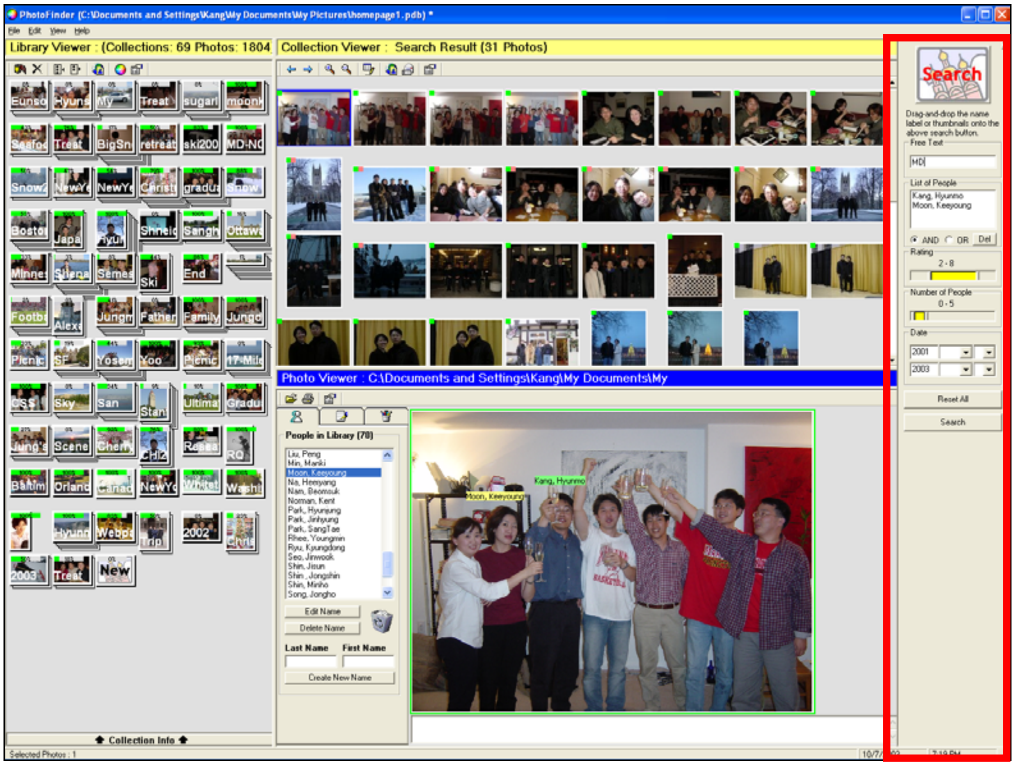


(d) Drag another name onto the search results in the CollectionViewer

Figure 3.1.5 Searching the annotated photos in PhotoFinder (Continued)


PhotoFinder (C:\Documents and Settings\Kang\My Documents\My Pictures\homepage1.pdb) *
Library Viewer : (Collections: 69 Photos: 1804
Collection Viewer : Search Result (31 Photos)
Photo Viewer : C:\Documents and Settings\Kang\My Documents\My
Search
Drag-and-drop the name label or thumbnails onto the above search button.
Free Text
List of People
AND OR Del
Rating
0 - 10
Number of People
0 - 30
Date
Reset All
Search
People in Library (70)
Edit Name
Delete Name
Last Name
First Name
Create New Name
Moon, Keeyoung
Collection Info

(e) Photos that do not contain the dragged name are filtered out from the previous search results in the CollectionViewer

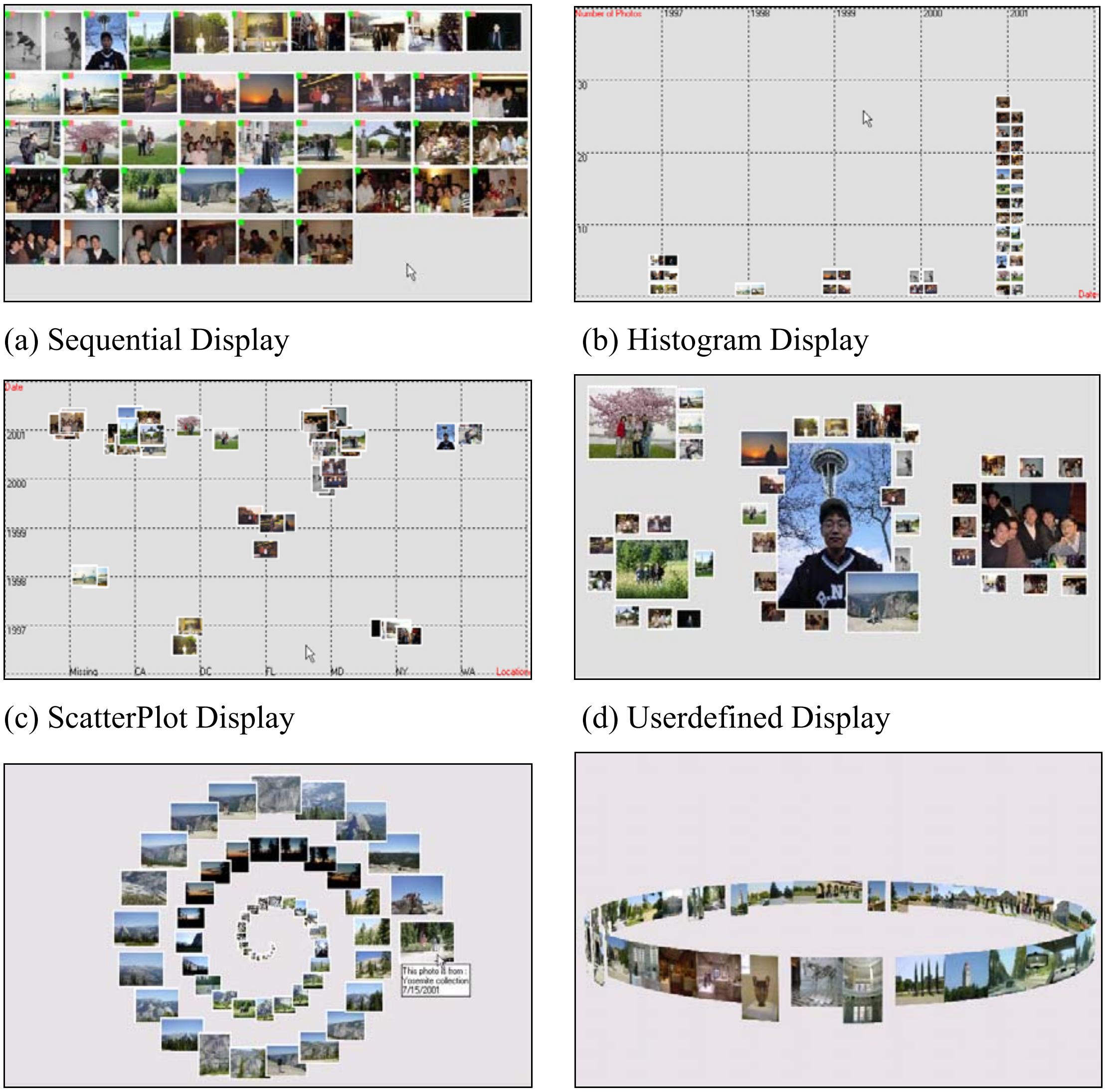

PhotoFinder (C:\Documents and Settings\Kang\My Documents\My Pictures\homepage1.pdb) *
Library Viewer : (Collections: 69 Photos: 1804
Collection Viewer : Search Result (31 Photos)
Photo Viewer : C:\Documents and Settings\Kang\My Documents\My
Search
Drag-and-drop the name label or thumbnails onto the above search button.
Free Text
MD|
List of People
Kang, Hyunmo
Moon, Keeyoung
AND OR Del
Rating
2 - 8
Number of People
0 - 5
Date
2001
2003
Reset All
Search
People in Library (70)
Kang, Hyunmo
Moon, Keeyoung
Collection Info

(f) More complex conjunctive queries are possible in the Search panel

Figure 3.1.5 Searching the annotated photos in PhotoFinder

The design of the collection viewer and the library viewer as well as the use of metadata in PhotoFinder motivated the concept of *Semantic Regions*. The collection viewer was designed to support several different display modes (**Figure 3.1.6**): sequential display, histogram display, scatterplot display, user defined display, spiral display and 3D merry-go-round display, in order to make it easy and fun for users to browse photos in a collection. Each display mode shows the different aspects of a photo collection by using photo metadata such as location, date, number of people, and rating. The sequential display places thumbnails one after another in a linear order without overlapping and enables users to sort them by filename or date (**Figure 3.1.6**(a)). The histogram display helps users to cluster photos based on an attribute and also makes it easy to recognize the amount of photos in each group with a histogram visualization (**Figure 3.1.6**(b)). In the scatterplot display, users can select two different attributes for x and y axes at once. Users can make use of this display for subcategorizing photos in a collection, or finding out any interesting correlations between the attributes (**Figure 3.1.6**(c)). User-defined display offers users a free space to arrange photos based on their story-telling threads (**Figure 3.1.6**(d)). Spiral display can be used to visualize the periodic time information for the photo search results (**Figure 3.1.6**(e)). 3D merry-go-round display gives users a perspective view of thumbnails with animation (**Figure 3.1.6**(f)).

In fact, multiple displays implemented in the Collection Viewer have little semantic information in their visual patterns since they make use of either standard mathematical/statistical visualizations (e.g. histogram and scatterplot) or geometric variations (e.g. spiral and circle). Even in this case, however, the different displays provide users with fast and insightful understanding of the photos in a collection. Furthermore, the automatic spatial organization of photos enables users to access photos very quickly, and thus expands users' capabilities of exploring the photos.



(a) Sequential Display

(b) Histogram Display

(c) ScatterPlot Display

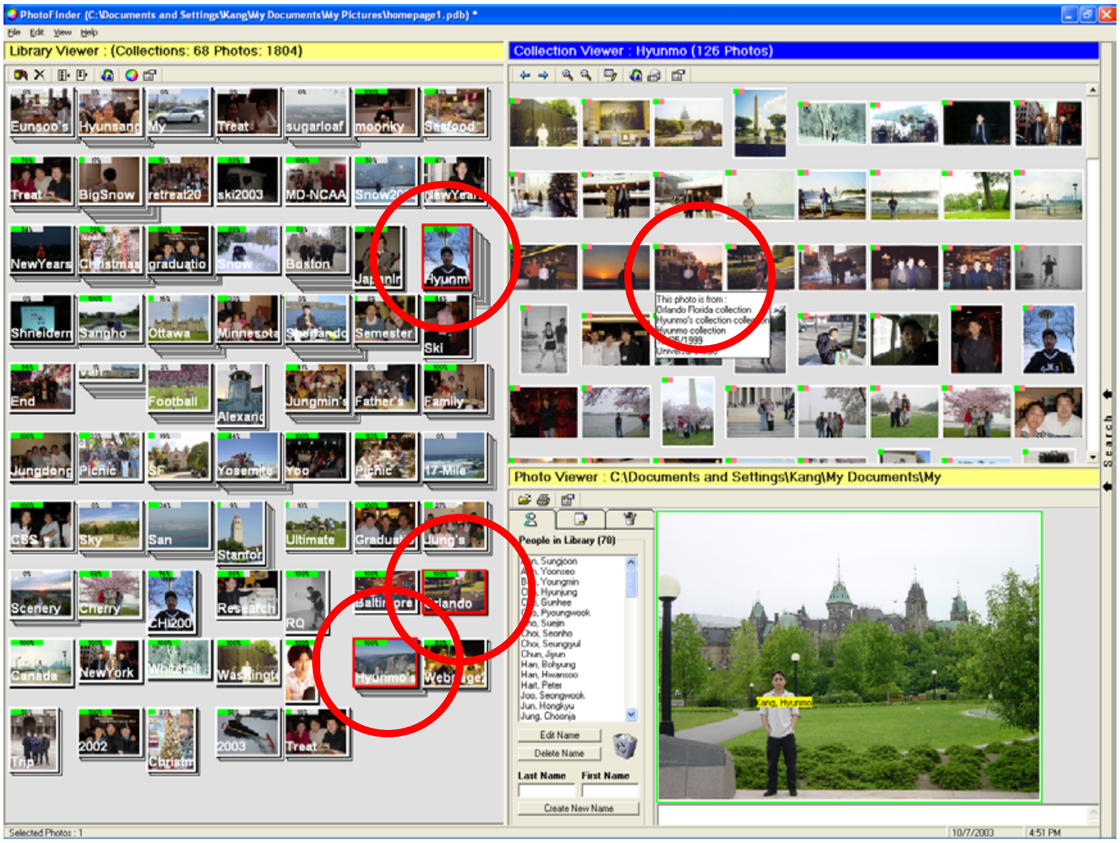
(d) Userdefined Display

(e) Spiral Display

(f) 3D merry-go-round Display

Figure 3.1.6 Different displays in the Collection Viewer

The collection viewer was also designed as a place for indexing and query formation. Users can select a group of photos in the Collection Viewer and assign an attribute value to them through bulk annotation. Simple queries such as "show the collection that contains the selected thumbnail" or "show detail information of the selected photo" can be performed just by placing the mouse cursor over a thumbnail (**Figure 3.1.7**). In short, the collection viewer is used as a basic user interface component for visualizing, annotating, and formulating queries in PhotoFinder.



PhotoFinder (C:\Documents and Settings\Kang\My Documents\My Pictures\homepage1.pdb) *

Library Viewer : (Collections: 68 Photos: 1804)

Collection Viewer : Hyunmo (126 Photos)

This photo is from :
Orlando Florida collection
Hyunmo's collection colle
Hyunmo collection
/05/1999
Universal

Photo Viewer : C:\Documents and Settings\Kang\My Documents\My

People in Library (70)

n, Sungjoon
A, Yoonseo
B, Youngmin
C, Hyunjung
C, Gunhee
o, Pyoungwook
o, Suejin
Choi, Seonho
Choi, Seungyul
Chun, Jiyun
Han, Bohyung
Han, Hwansoo
Hart, Peter
Joo, Seongwook
Jun, Hongkyu
Jung, Choonja

Selected Photos : 1

10/7/2003

4:51 PM

Figure 3.1.7 When a mouse is over a thumbnail in CollectionViewer, a tooltip shows the information of the selected photo (query preview) and all the collections that contain the selected photo are highlighted red (brushing)

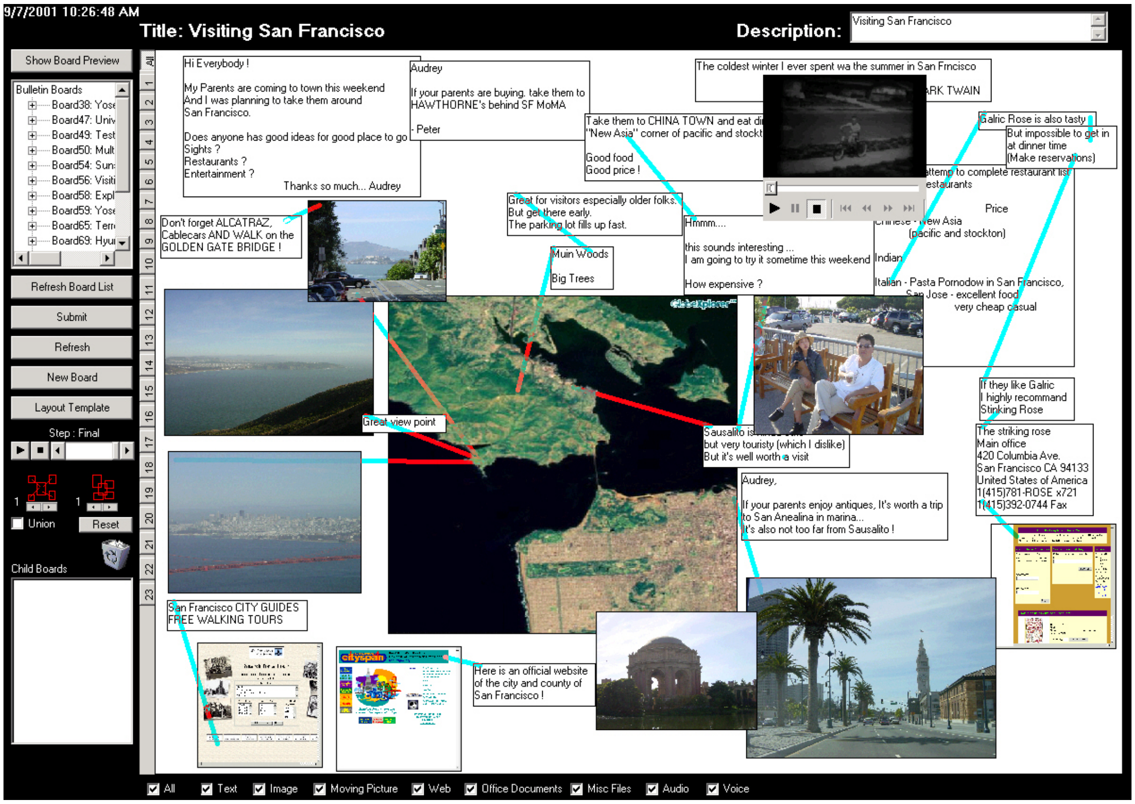
On the other hand, the library viewer enables users to classify the photos into several collections according to users' categorizing strategies (e.g., collection by events, collection by person, collection by time, and so on). Even though the photo collections in a library could not be regrouped dynamically in PhotoFinder3, the library viewer allows users to apply their own personal ontologies toward the photo data set. In addition, allowing a photo to be contained in multiple collections enables more flexible photo organizations compared to the current file folder hierarchy metaphor.

3.2 Multimedia Bulletin Board

Another project "Multimedia Bulletin Board", which the author worked on during the summer of 2001 at Ricoh California Research Lab, was also a major motivating factor for engendering the concept of S*emantic Regions* in terms of applying various mental models by restructuring the layout of the multimedia objects.

The *Multimedia Bulletin Board* (*MBB*) was designed and implemented as a prototype of the next generation asynchronous information system that enables rich communication and collaboration among people using multimedia objects such as text, image, moving picture, sound, voice, web, document, and miscellaneous files. The main *MBB* display window with a control bar (**Figure 3.2.1**) is implemented as an ActiveX control and embedded in a web page so that users can easily manipulate the boards with web browsers.

Figure 3.2.1 shows an example of the *MBB* messages on the topic of "Visiting San Francisco" which was composed by multiple users. One user triggered the discussion by asking for some good places to visit, such as restaurants and clubs. Several users answered the question and made comments on the posted messages with maps, images, web pages, moving pictures, texts, and so on. The layout properties of the multimedia objects such as x-y position, size, z-order, partial occlusion, explicit links, spatial distances among the objects, etc. represent useful and important information on the user-dynamics related to the specific story threads or the discussion topics on the board. However, the fixed layout (manually created by multiple users) may prevent users from extracting and identifying information. For example, users might be interested in such tasks as “I’d like to see the linear list of restaurants that people recommended with their related web pages and pictures”, “I’d like to see all the landscape pictures posted, placing them around the San Francisco area map with reasonable size without overlapping”, “I’d like to see all the messages posted on this board in chronological order using a weekly calendar”, “I’d like to see how two specific people interacted with each other in this board” and so on. There is no easy way to perform these tasks with the fixed layout of a board, and even if the layout is not fixed, it is a tedious and time-consuming job for users to change the whole layout manually every time they want to apply their own mental models to a board.


9/7/2001 10:26:48 AM

Title: Visiting San Francisco

Description: Visiting San Francisco

Hi Everybody !

My Parents are coming to town this weekend
And I was planning to take them around
San Francisco.

Does anyone has good ideas for good place to go
Sights ?
Restaurants ?
Entertainment ?
Thanks so much... Audrey

Audrey

If your parents are buying. take them to
HAWTHORNE's behind SF MoMA

- Peter

The coldest winter I ever spent wa the summer in San Frncisco

ARK TWAIN

Take them to CHINA TOWN and eat di
"New Asia" corner of pacific and stockt

Good food
Good price !

Galric Rose is also tasty

But impossible to get in
at dinner time
(Make reservations)

attemp to complete restaurant list
estaurants

Price

Chinese - New Asia
(pacific and stockton)

Indian

Italian - Pasta Pornodow in San Francisco,
San Jose - excellent food
very cheap casual

Great for visitors especially older folks.
But get there early.
The parking lot fills up fast.

Muin Woods

Big Trees

Hmmm....

this sounds interesting ...
I am going to try it sometime this weekend

How expensive ?

Don't forget ALCATRAZ,
Cablecars AND WALK on the
GOLDEN GATE BRIDGE !

Great view point

Sausalito is
but very touristy (which I dislike)
But it's well worth a visit

If they like Galric
I highly recommand
Stinking Rose

The striking rose
Main office
420 Columbia Ave.
San Francisco CA 94133
United States of America
1(415)781-ROSE x721
1(415)392-0744 Fax

Audrey,

If your parents enjoy antiques, It's worth a trip
to San Anealina in marina...
It's also not too far from Sausalito !

San Francisco CITY GUIDES
FREE WALKING TOURS

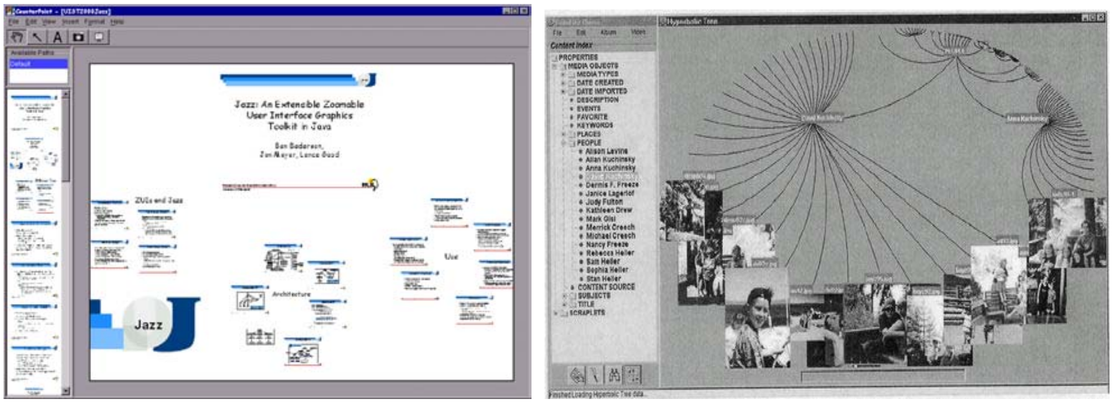
Here is an official website
of the city and county of
San Francisco !

Bulletin Boards
Board38: Yose
Board47: Univ
Board49: Test
Board50: Mult
Board54: Sun
Board56: Visiti
Board58: Expl
Board59: Yose
Board65: Terr
Board69: Hyu

Figure 3.2.1 Multimedia Bulletin Board message on the topic of “Visiting San Francisco”

In practice, users create meaning by organizing the space [54]. There are many ways to lay out the same set of messages. There is no “right” layout. Rather, the layout is very personal and has meaning for the individual who creates it. Therefore, the main challenge was to design and implement a novel user interface that makes it easier for end-users to restructure the layout of existing multimedia messages on the board and create new layouts of messages. As the number of the messages on the board increases, the need and benefit of restructuring the layout of the board grows.

With email lists, users expect to be able to reorganize their linear textual displays in ascending/descending order according to date received, sender name, topic, size, etc. With multimedia information, designers are beginning to understand what forms of two-dimensional displays are needed and explore novel possibilities. For example, MS PowerPoint allows only a row-by-column sequential ordering of slide thumbnails, but newer tools, such as Counterpoint [26] enable richer possibilities (**Figure 3.2.2**(a)). Counterpoint users can have circular, oval, or hierarchical displays that reveal the structure of a presentation and allow greater flexibility in preparing and presenting a slide show. Photo browsers are also expanding from row-by-column displays that are found in ACDSee (http://www.acdsystems.com), PhotoSuite (http://www.mgisoft.com), and web tools such as Ofoto (http://www.ofoto.com). PhotoFinder [33][61] offers several collection viewers that allow user control of size and placement or automatic placement by ranking, date, numbers of people, etc. Canon's ZoomBrowser offers a dazzling spiral display that is attractive to many viewers and the FotoFile [35] uses a hyperbolic tree for photos (**Figure 3.2.2**(b)). PhotoMesa is a zoomable image browser [6]. It allows the user to view multiple directories of images in a zoomable environment, and also supports grouping of images by metadata available from the file system such as directory location, image creation date, and words in the filename. PhotoMesa introduces two layout algorithms - Quantum Treemaps and BubbleMaps to layout the images on the screen, using space efficiently while showing the groups the images are in (**Figure 3.2.3**). Other innovations include 3-dimensional positioning of photos [36] and user fly-throughs that are appealing for some users, but disorienting for others.


Jazz: An Extensible Zoomable User Interface Graphics Toolkit in Java
Ben Bederson, Jon Meyer, Lance Good
ZUIs and Jazz
Architecture
Use
Jazz
Hyperbolic Tree

(a) In CounterPoint, a presenter can organize slides into a hierarchy and apply layout templates to automatically arrange slides along predefined shapes

(b) In FotoFile, photo thumbnails can be arranged with a hyperbolic tree for photo browsing

Figure 3.2.2 The example applications of spatial layout

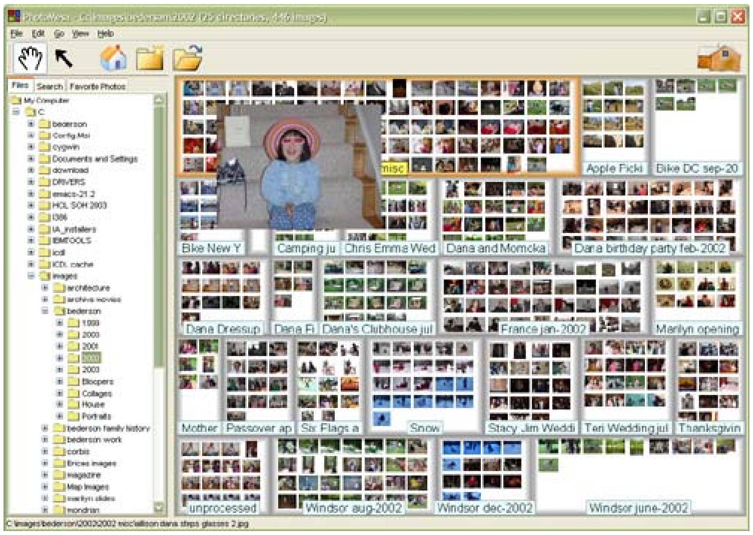
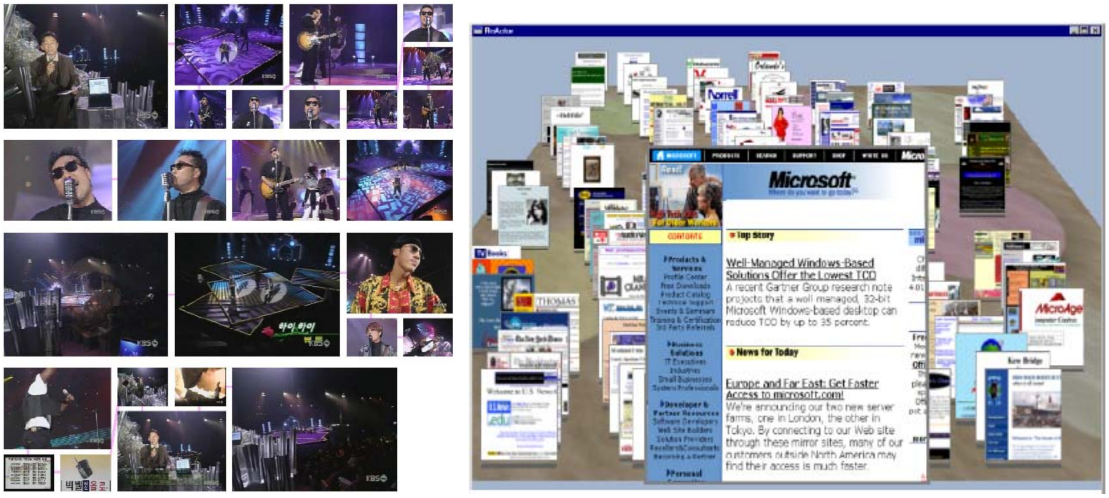


Figure 3.2.3 PhotoMesa: Spatial layout of photos using Quantum Treemaps

Some studies focused on the model for window management and layout in desktop interfaces. Elastic windows [31] presents a browsing interface with a hierarchical window organization and multiple window operations that allows users to organize web pages and restructure the information on the screen. Bell and Feiner [7] present a general approach to the dynamic representation of 2D space that is well suited for tiled user interface layout. Beaudouin-Lafon [4] presents several techniques that improve window management by extending the metaphor of overlapping windows. However, most researchers have addressed window and space management to maximize visibility and usability rather than constructing new meaning through dynamic layouts of the content objects.



Well-Managed Windows-Based Solutions Offer the Lowest TCO
A recent Gartner Group research note projects that a well managed, 32-bit Microsoft Windows-based desktop can reduce TCO by up to 35 percent.

Europe and Far East: Get Faster Access to microsoft.com!
We're announcing our two new server farms, one in London, the other in Tokyo. By connecting to our Web site through these mirror sites, many of our customers outside North America may find their access is much faster.

(a)Video Manga: comic book style video summary (b) Data Mountain
Figure 3.2.4 Meaning construction using spatial layout of information

Video Manga [72] shows a good example of constructing new meaning through spatial layout (**Figure 3.2.4**(a)). It presents methods for automatically creating pictorial video summaries that resemble comic books. The selected keyframes of each video segment are sized by importance and then packed into a pictorial summary. Microsoft's Data Mountain [54] shows another possibility for using spatial layout for document management. In this system, users freely arrange document thumbnails on an inclined 2.5D plane textured with passive landmarks to take advantage of human spatial memory (**Figure 3.2.4**(b)).

There were several design issues concerning the type of objects, interaction metaphor, history, filtering, navigation, board hierarchy to enrich communication, and collaboration among people. The multimedia objects supported by *MBB* can be classified into three categories; visual object, audio object, and file/URL object. The visual object category contains text objects, image objects, and moving picture objects. Users can type in text strings on the board to make annotations of other objects or just for displaying the text messages. Users can also select and drag the text strings from other applications (e.g. word processors, web browsers, email clients, etc.) onto the board for posting the information. Similarly, image objects and moving picture objects can be dragged onto the board from the file browsers or other image browser/moving picture applications. Users can freely move and resize the visual objects on the board.

The audio objects include sound and voice objects. Voice objects record the users' voice and generate an audio file that can be used for annotating other objects or for posting voice messages. Sound objects are used for posting built in audio files on the board. Users can play voice and sound objects without any audio player applications installed on their machines. The File/URL category is used for transferring files or accessing the URLs through *MBB*. This category contains web objects, office document objects and miscellaneous file objects. When users drag a URL from a web browser onto the board, the *MBB* client automatically generates the thumbnail of the web page and places it on the board as a web object. This thumbnail approach gives users more visible information about the posted URL [77]. The URL and title of a web page are shown as a tooltip text. The web object is handled just like an image object, however the designated action on this object is to launch a web browser and load the URL specified in it.

The *MBB* can also be used as a storage archive system with the miscellaneous file objects. Users can upload any type of file by dragging it from a file browser onto the board. In addition, users can download a file from the board by double clicking on a miscellaneous file object. The office document object is treated slightly different from the miscellaneous file object. This is because users are using office documents everyday and the visual displays of the office documents help users identify them more easily. Like a web object, the *MBB* client automatically generates the thumbnail image of the office document when users drag an office document onto the board.

Simple drag-and-drop interaction is used as a basic metaphor for composing messages such as add, delete, move, resize, link, and so on. This was intended to minimize users' difficulties with managing and posting the objects on the board. In addition, the *MBB* was designed to perform a proper action based on the object type such as "launch the associated applications" or "download the multimedia files" for the users' double-click interaction on the multimedia objects. Like other asynchronous communication systems, all the changes on the board are updated and sent to the server only after users press the submit button. When the button is pressed, a new transparent layer is added on top of the current board.

Since an actual bulletin board only shows its final (current) state, it is difficult for users to identify the previous state of the board or to observe the evolving sequence of the board. This information may be useful for recognizing the user dynamics on the specific story thread or the discussion topic. The history keeping mechanism implemented in *MBB* enables users to go back to any previous state of the board. It also makes it possible for users to play back a specific range of the board states.

It is not unusual that several different story threads coexist within a board. If the board is crowded with other messages, it is difficult for users to follow the messages along the specific story thread or to recognize only the messages they are interested in. The *MBB* supports several filtering functions to resolve the problems. From the observation on the usage of an actual bulletin board we found two interesting facts, which influenced the design of the filtering functions. The first fact is that users have a tendency to make an explicit link (line or arrow) between the objects to make their messages belong to the specific story thread. The second one is that users tend to keep their messages as close as possible to the related message in case they don't make an explicit link. Since users depend on the spatial locality among the related messages, we define it as an implicit link. Users can filter out objects on the board by object type, author, layer, and creation time. Users can also filter out objects by specifying the depth of an explicit or implicit link path from the selected object. The filtered sets of the messages can be combined conjunctively and the history keeping mechanism can be applied to the filtered objects as well.

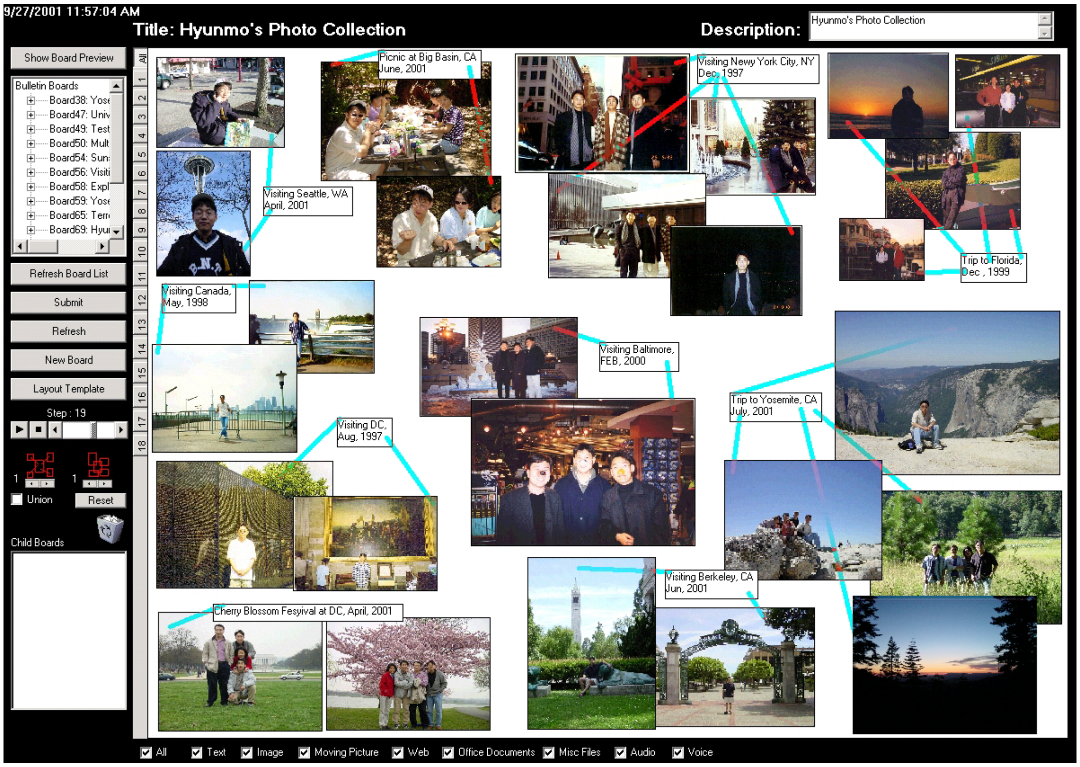
Because the initial *MBB* prototype was designed to have a fixed size display (1024*768 pixels) without any navigation mechanism, users need some way to expand board space without losing the context of the existing board. To resolve this problem, *MBB* allows a hierarchical relationship among the boards. If there is not enough space to add a new multimedia object on the board, users can shrink the current board into a new image object and add it to a new board. The newly created board is inserted as a child of the current board and users can easily go back to the parent board by double clicking on the board image object. Users can preview the children of the current board as thumbnails in the *MBB* control bar, and they can also browse the whole hierarchy of the boards with the treeview list. The operations of creating a child board and going back to the parent board are performed with animation to minimize users' confusion and frustration which may arise from jumping to another board in the hierarchy.

These novel features in the *MBB* prototype enabled users to have richer communication and collaboration, but there still remained many challenges. Most features mentioned above were designed and implemented mainly for helping users browse, filter, search, and follow the threads of messages on the board as they were initially created. However, in many cases, users need to perform more sophisticated tasks that cannot be easily achieved by keeping the existing structure of the messages. The partial or entire occlusion of messages often prevents users from identifying and recognizing the information on the board. Therefore, users need a way to manipulate the message content and display it in a clear way.

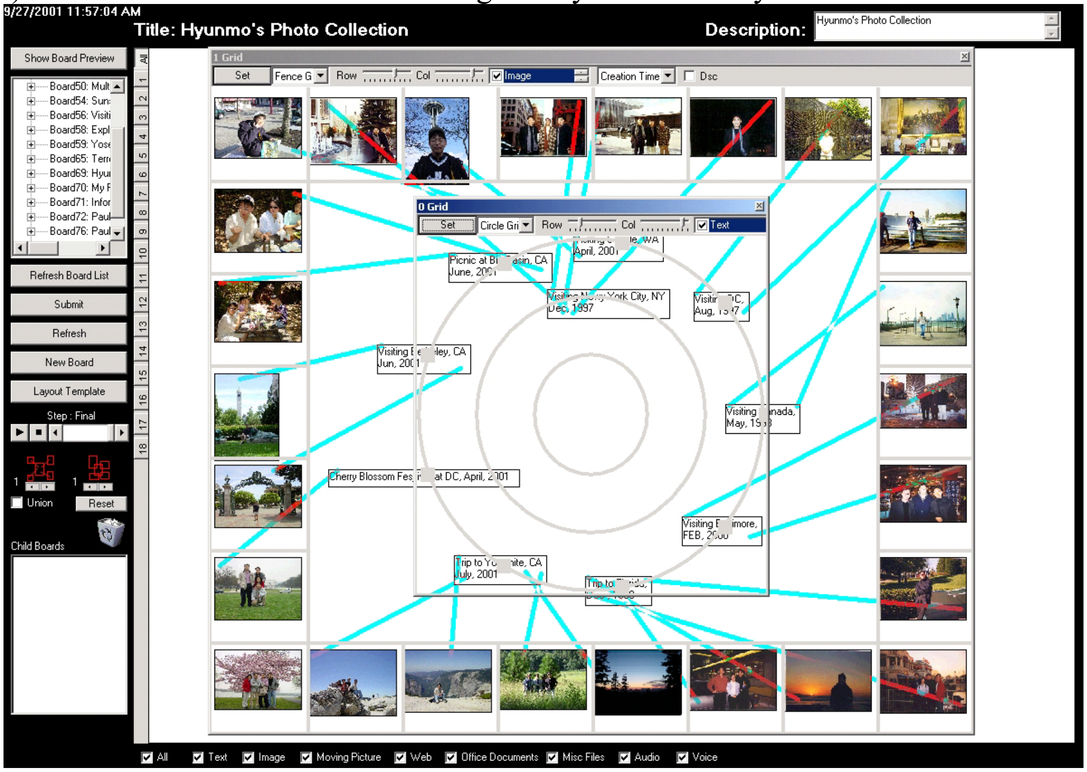
To cope with these challenges we developed the concept of *dynamic layout template* (*DLT*) to dynamically restructure the layout of the multimedia messages on the board. In *MBB*, all the multimedia objects and their relations on the board are represented as a graph structure. Each of the multimedia objects represents a node while a relation between the multimedia objects represents a link. The flow algorithm defined in the *MBB* projects the internal data representation (graph) onto the 2D space (board). *DLT* enables users to restructure any part of a board layout dynamically by applying a new flow algorithm to the designated part of the internal data representation.

MBB users can choose any number of *DLT*s simultaneously and apply them to the designated subset of the messages on the board. Users can select the subset of the message objects by object types, authors, time period, etc. and they can also select the objects by specifying a rectangular region on the board. *DLT*s were designed and implemented as resizable see-through windows to maximize the visibility of the objects and the flexibility of object management. In order to minimize users' confusion and frustration which may arise from the sudden layout change, all the layout transformations performed by *DLT*s are animated. All the layout transformations are reversible and recorded in the history on the client machine.

In the initial prototype, five sample *DLTs* were implemented: grid (2D row by column), fence-grid (2D row by column but using only outside rows and columns), calendar (shows a month organized by weeks), circular (objects are evenly spaced in a circle that fits the board), and spring model (circular with a central image and no overlapping links). **Figure 3.2.5**(a) shows the sample personal photo collection board created by three users. Image objects with captions (text objects) on the board are clustered into a few groups according to the related events by end-users. **Figure 3.2.5**(b) shows the transformed board layout after two *DLTs* (fence-grid and circular *DLTs*) have been applied to each type (image and text) of the multimedia objects on the board. Fence grid *DLT* is the layout template that places the grid cells along the four sides (top, right, bottom, and left) of the window. Users can specify the number of the horizontal and vertical cells in it. Circular *DLT* places the objects along the circumference of the circles. Users can select the number of concentric circles (they can be spaced evenly or relatively to the radius of circles) and the number of points where the objects will be placed. In Figure **Figure 3.2.5**(b), image objects are sorted by creation time and arranged clockwise around the board with the same size. Similarly, text objects have been arranged along the circumference of the circular *DLT* that is placed inside the fence-grid *DLT*. With this newly formed layout, users can see the image objects more clearly in chronological order without any occlusions. Users can assign a different size to each of the multimedia objects according to any of the attributes stored in the *MBB* database. For example, users can resize each of the selected image objects relative to the initial image size or in chronological order through the *DLT*. **Figure 3.2.6**(a) shows the sample web bookmarks board created by a single user. The user dragged some of the interesting URLs from the history of a web browser onto the board to save them as his favorites or bookmarks. The automatically generated web objects are placed on the board according to the user's grouping mechanism just as in Microsoft's Data Mountain [54]. **Figure 3.2.6**(b) shows the newly generated layout of the board by applying two calendar *DLTs* (Jan 2002 and Feb 2002). All the web objects are arranged inside the calendar *DLTs*. If more than one web object is placed within a grid cell, users can choose either a tile (no occlusion but different size) or cascade (allow occlusions but equal size) arrangement. In this example, the upper left calendar *DLT* uses the cascade arrangement and lower right *DLT* uses the tile arrangement.

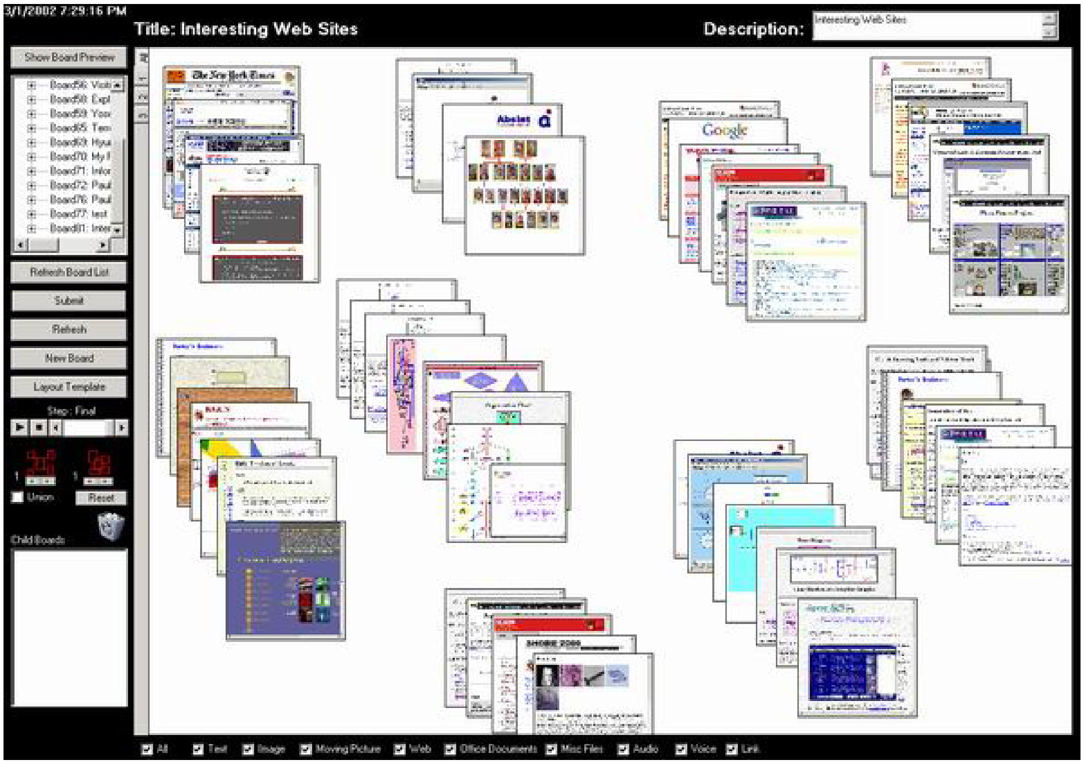


(a) Photo collection board with the original layout created by three users



(b) Image objects are sorted and arranged by fence-grid *DLT,* and text objects by circular *DLT*.

Figure 3.2.5 Example dynamic layout management through *DLT*

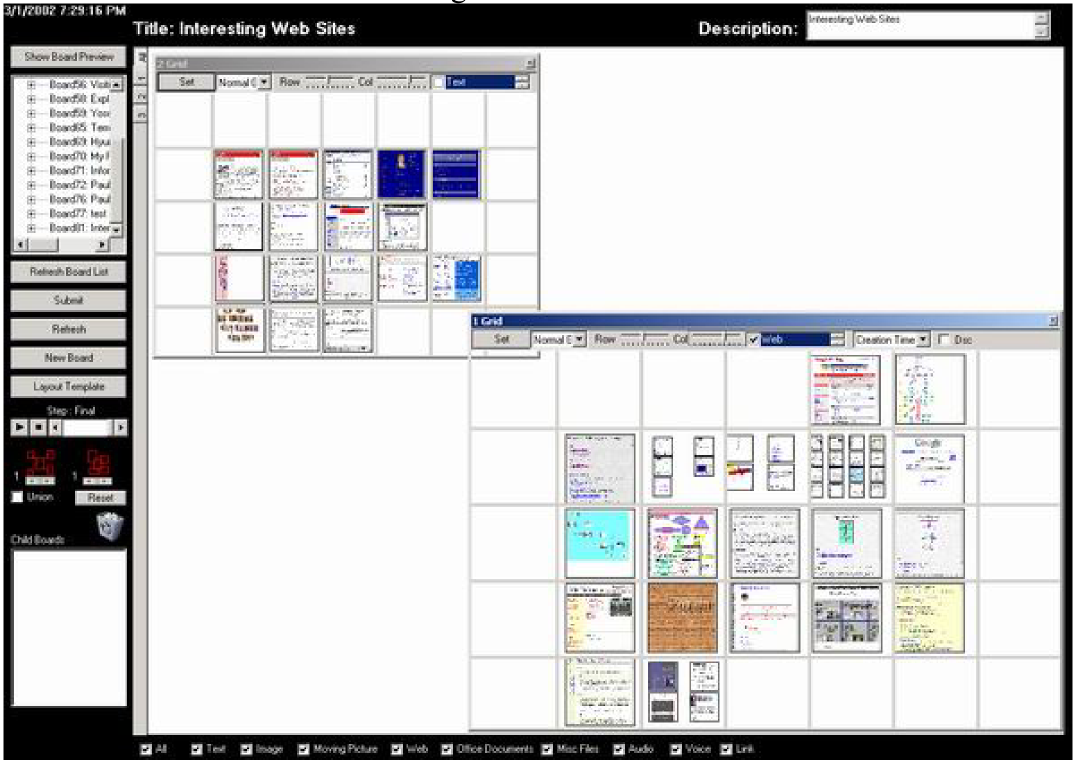


3/1/2002 7:29:16 PM

Title: Interesting Web Sites

Description: Interesting Web Sites

Figure 3.2.6 (a) Personal web bookmarks with the original layout created by a single user. Similar to Data Mountain arrangement with occlusions.



3/1/2002 7:29:16 PM

Title: Interesting Web Sites

Description: Interesting Web Sites

Figure 3.2.5 (b) Web objects are sorted and rearranged by two different size calendar *DLTs.* Bottom right *DLT* uses the tile arrangement to avoid occlusions.

In summary, even though there were many challenges and issues to be revised in this research, the concept of *Dynamic Layout Template* in *Multimedia Bulletin Board* showed an initial prototype of applying various mental models to personal media data.

The application of multiple *DLT*s to the messages on a board led to the idea of applying multiple mental models to personal media to observe the interrelationship between the different models. Also, the idea of specifying the rules to *DLT*s for choosing a subset of multimedia messages on the board resulted in the development of the concept of specifying the semantics for each *Semantic Region.* In addition, future work for supporting a tool that enables users to construct user-defined *DLT*s affected the development of MediaFinder with which users can construct their own mental models for personal media data. Spatial organization of multimedia messages on a board also affected the current research of *Semantic Regions*.

3.3 Summary

The studies on the preliminary personal media management systems led to a new interface design paradigm and an innovative concept for managing personal media data, *Semantic Regions*.

The PhotoFinder project shows the importance of the use of semantics in managing the personal photo library and suggests ways to use semantics for searching and browsing. The needs for user-defined metadata led to the development of a novel manual indexing technique called *direct annotation.* This technique has been extended to annotate a group of photos with one simple action called *bulk annotation*. In addition, multiple displays designed in CollectionViewer and features of spatially organizing photos on 2D space expand users' capabilities of exploring photos. Furthermore, the design of the Collection viewer as a basic GUI component for indexing, visualization and query formation led to an initial prototype of a multifunctional GUI component, *Semantic Regions*. Also, the design of Library Viewer enables users to organize their photos freely based on their mental models or specific tasks so that they need not adhere to the system-oriented file-folder hierarchy model.

On the other hand, the Multimedia Bulletin Board project shows the effectiveness of spatial organization of multimedia objects on 2D space. Moreover, the specification of the rules to *DLT*s for grouping the multimedia objects that satisfies the rules led to the idea of dynamic regrouping of personal media objects based on their semantics in *Semantic Regions.* Also, applying multiple *DLT*s to the multimedia objects on a board led to the idea of combining multiple mental models to observe the interrelationship among them. Finally, the idea of constructing user-defined *DLT*s affected the development of MediaFinder with which users can construct their own mental models for personal media data.

Chapter 4:

MediaFinder: User Interface for Semantic Regions Construction and Operation

4.1 *Semantic Regions*: Interactive Personal Media Query Widget

Semantic Regions are query regions drawn directly on a two-dimensional information space. Users can specify the shapes, sizes, and positions of the regions in two-dimensional space and thus form the layout of the regions meaningful to them. Once the *Semantic Regions* are created and arranged based on the users' mental models, users can specify semantics for each region. However, there is no restriction on the order of creating, arranging, and assigning the semantics to the regions. In other words, users can assign and modify the semantics of each region while they are creating and arranging it. The semantics of a region consist of file system attributes and user-defined high level attributes. The attributes are conjunctively joined to form the semantics of a region.

A *Semantic Region* works just like a folder. Multiple personal media items are contained in a *Semantic Region* just as multiple files are contained in a folder. *Semantic Regions* can have hierarchies. Child regions can be defined inside a parent region like subfolders. However, there are two major differences between a *Semantic Region* and a folder. First, a *Semantic Region* has its semantics and it only contains the personal media items that satisfy the semantics. A *Semantic Region* automatically gathers the personal media items from the bound data based on its semantics, and users cannot enforce the personal media data to be placed in a region manually unless the attributes of personal media items are changed. Second, a personal media item can be contained in multiple regions as long as it satisfies their semantics. Unlike the file-folder structure, *Semantic Regions* need not to be mutually exclusive, and thus enable more flexible organizations of personal media data. Other detailed differences are explored in the following sections.

In summary*, Semantic Regions* are basic interactive query widgets and their integration forms a model for personal media management such as organization, meaning extraction, navigation, search, indexing, and distribution.

4.2 MediaFinder

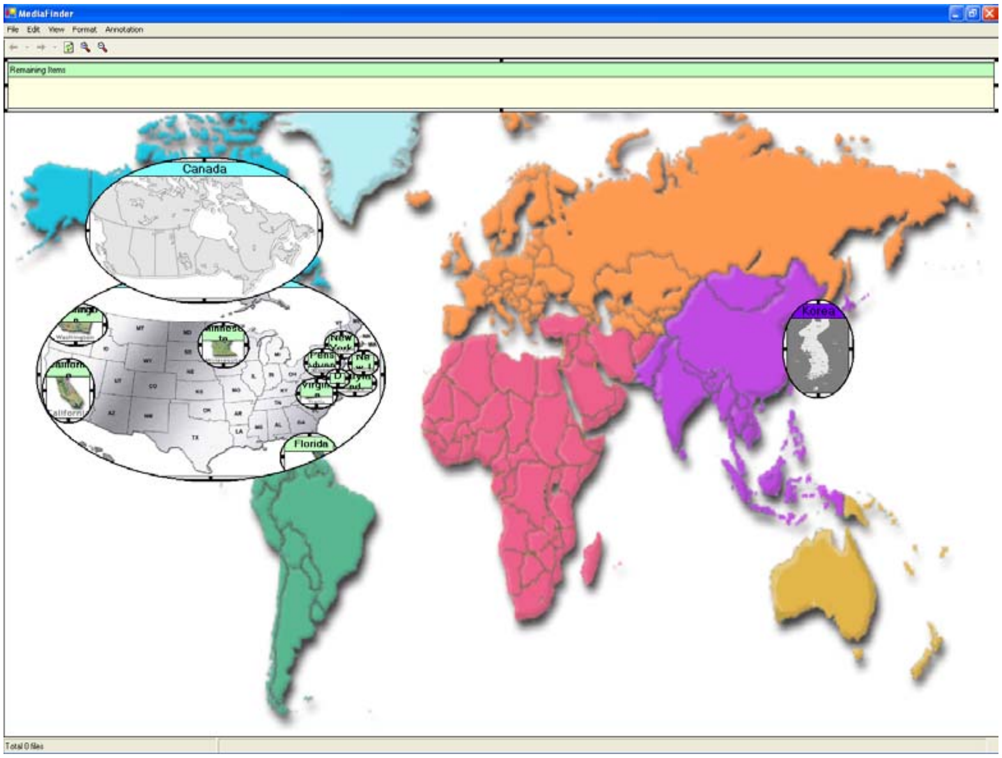
The *MediaFinder* is a prototype interactive tool built to investigate the use of S*emantic Regions* for personal media management and exploration. MediaFinder provides a working environment for *Semantic Regions* construction and operation. Also, MediaFinder functions as a repository to hold a domain personal media data set for data binding. Users can bind personal media data statically or dynamically by mapping a folder to MediaFinder or by *fling-and-flock*. In addition, MediaFinder provides users a logical working space to represent their mental models, combine them, and use them for personal media management and exploration.

The complete MediaFinder user interface for *Semantic Regions* construction and operation is described in the following sections. As a future work, the initial MediaFinder prototype will be extended to manage the entire heterogeneous personal media data stored in users' individual machines, and it will also be extended to support a database system by providing visual queries.

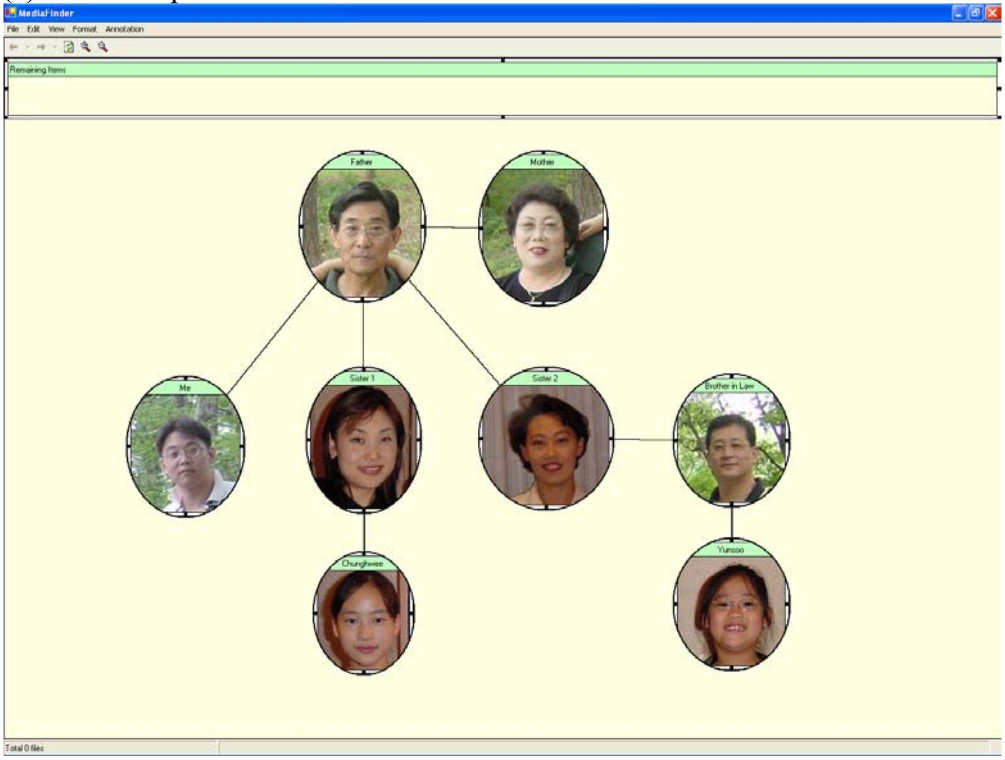
4.3 Constructing Semantic Regions in Media Finder

MediaFinder is composed of a two-dimensional media space (workspace) and many functions that help users to construct and use *Semantic Regions*. In the workspace, users can freely create, remove, resize, move, and change the z-order of *Semantic Regions* just like an object-oriented drawing tool. Several drawing and formatting features neatly implemented in MediaFinder enable users to easily outline their mental models on the 2D media space. After users sketch their mental models, they need to specify the semantics for each region to represent their meanings.

Figure 4.3.1 shows a few example models (world map, genealogy tree, season calendar, and group organization chart) of *Semantic Regions* created with MediaFinder. *Semantic Regions* are spatially positioned and grouped on the 2D space according to well known display representations such as a map, tree, timeline, or org chart. The background images of the MediaFinder main window and the *Semantic Regions* are used to help users understand the models easily. Some models were created to have hierarchies. For instance, the world map has country regions, the US region has the state regions, and a state region has the city regions (**Figure 4.3.1**(a)). Each region has its own semantics. For example, the country name is specified for each country region in the world map model, a person's name is specified for each person region in the genealogy tree model, and the date period of the season is specified for each season region in the seasonal calendar model. The semantics can be conjunctively joined to imply more complicated meanings.

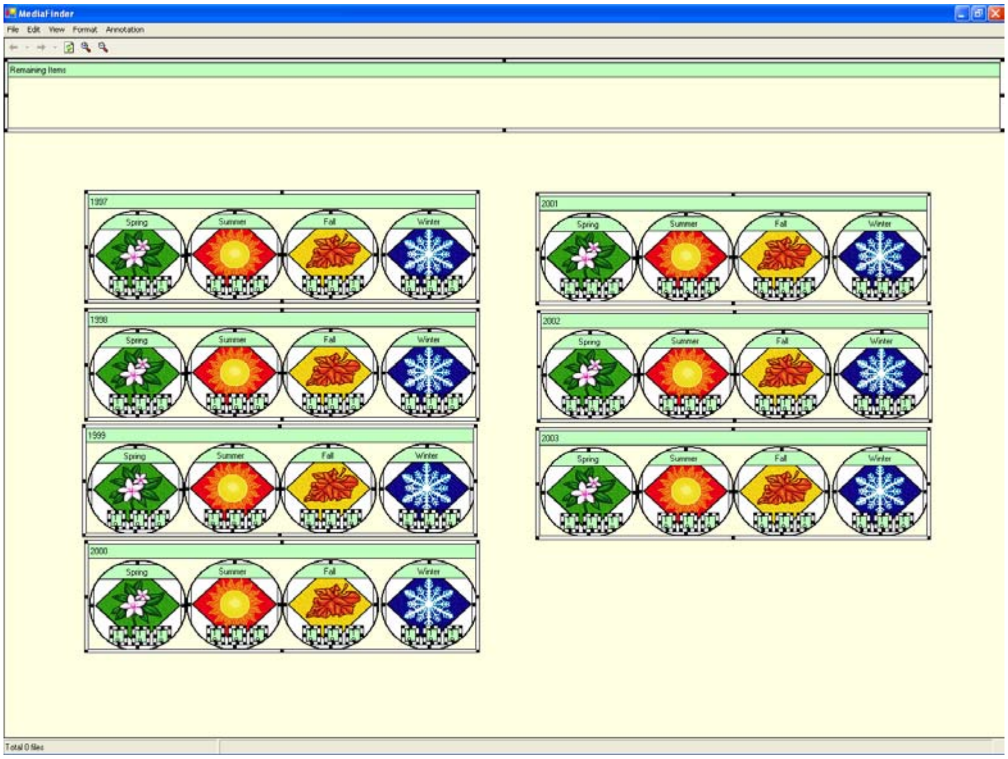


(a) World map model

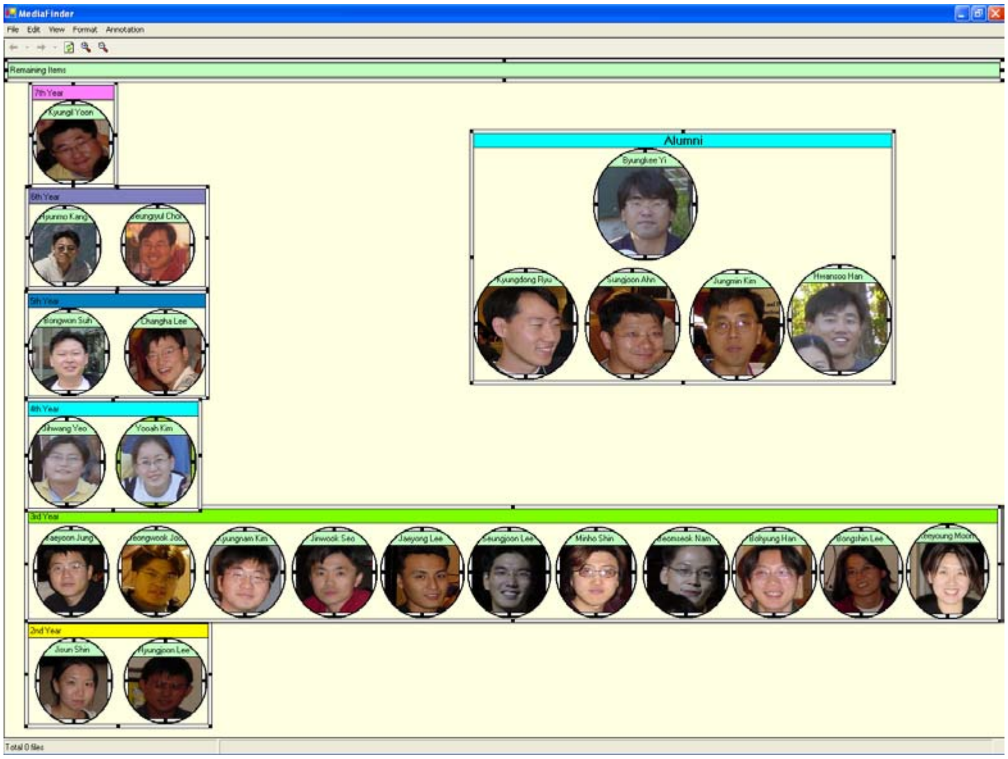


(b) Genealogy tree model

Figure 4.3.1 Example models of *Semantic Regions* constructed with MediaFinder (Continued)



(c) Season calendar model

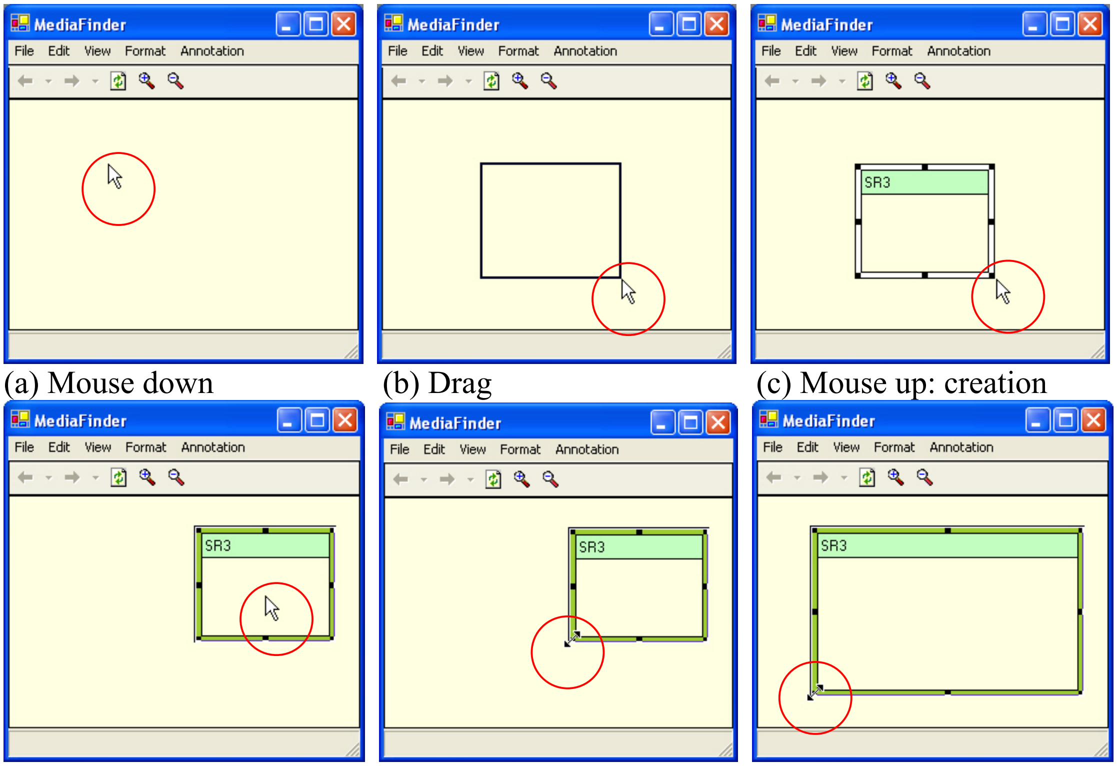


(d) Group organization chart model

Figure 4.3.1 Example models of *Semantic Regions* constructed with MediaFinder

4.3.1 Semantic Region Construction

The creation of *Semantic Regions* is straightforward: users simply click on the desired starting point of the semantic region and drag the pointer to the desired location of the opposite corner. As this is identical to the mechanism used for creating rectangles in widely used drawing programs, this operation should be familiar to most users. Once the semantic region is created, it may be dragged to a new position or resized via appropriate resize handles on the corners, using similarly familiar interactions (**Figure 4.3.2**).



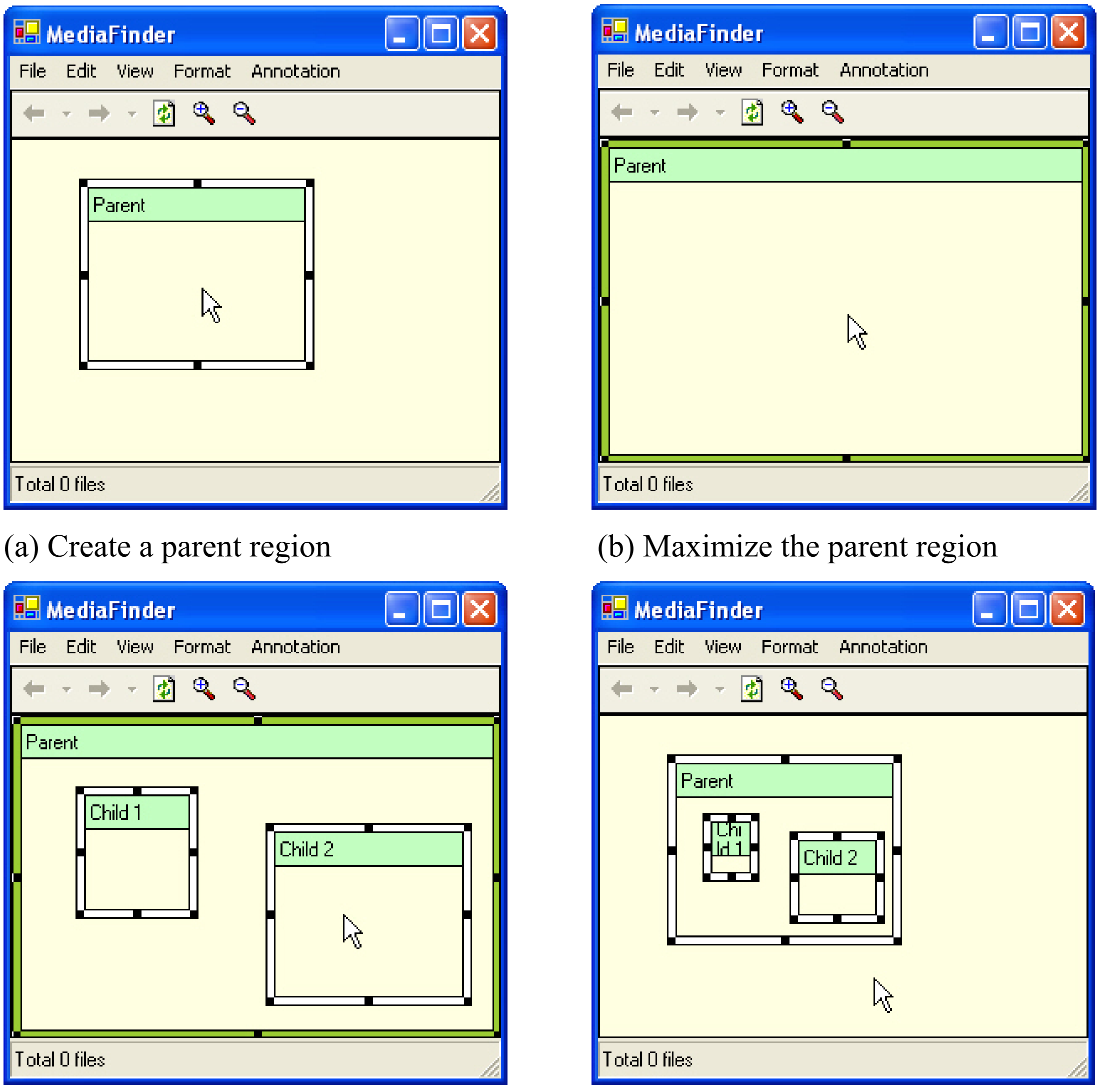
(a) Mouse down (b) Drag (c) Mouse up: creation

(d) Select and move (e) Drag a corner (f) Resize

Figure 4.3.2 Creating, moving, and resizing *Semantic Regions* by Click-and-Drag

Users can change the shape of the selected regions or add background images to clearly represent the implication of the region and thus easily remember it for later use. Users can also specify a title for a region for the same purpose.

The child *Semantic Regions* can be constructed recursively inside a region (**Figure 4.3.3**). The background of each region is used as a new 2D workspace on which new child regions are created. The semantics of the child regions are applied only to the personal media items that are contained in their parent region. Hence, if the semantics of the parent region are modified, then the contents (personal media items contained in the region) of the child regions are dynamically changed.



(a) Create a parent region

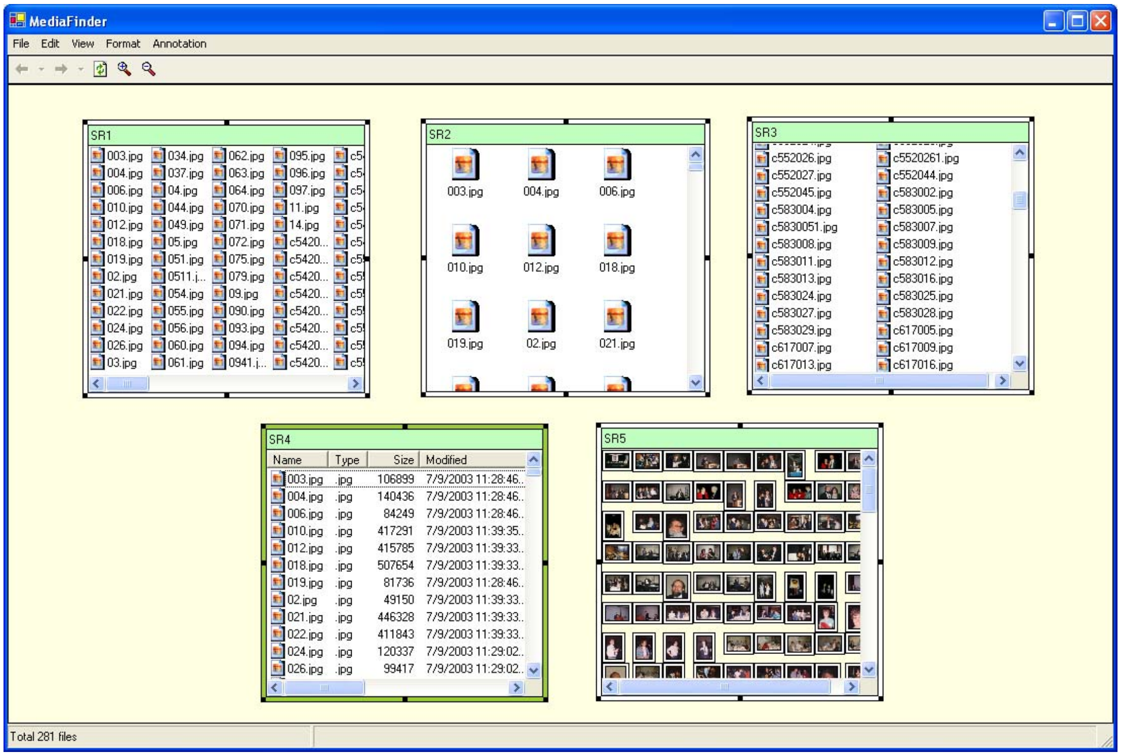
(b) Maximize the parent region

(c) Create child regions inside the parent

(d) Minimize the parent region

Figure 4.3.3 Creating child regions inside a parent region

Each region has multiple views (**Figure 4.3.4**) to provide various perspectives of the contained personal media item. Each region supports five basic views (list, small icon, large icon, details, and thumbnails) just as the Windows file explorer does. Users can launch the associated applications by double-clicking over a personal media item in these views just as they can do with Windows file explorer.

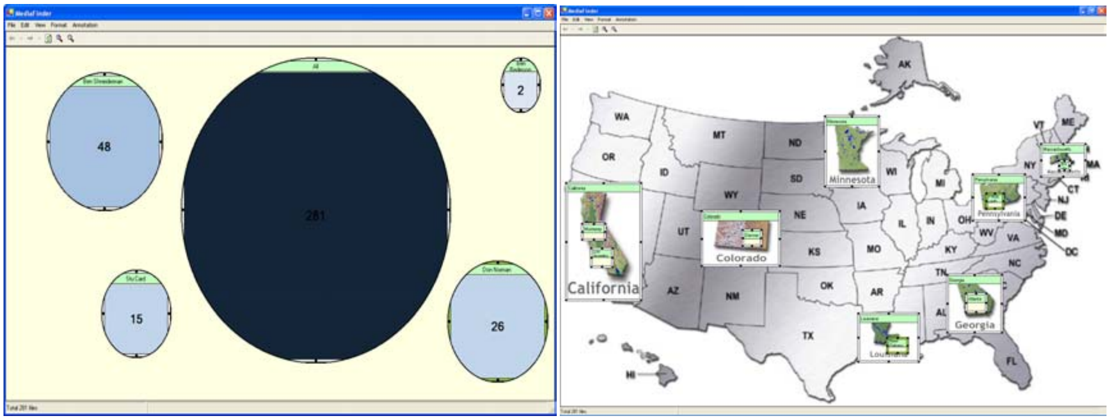


Name	Type	Size	Modified
003.jpg	.jpg	106899	7/9/2003 11:28:46..
004.jpg	.jpg	140436	7/9/2003 11:28:46..
006.jpg	.jpg	84249	7/9/2003 11:28:46..
010.jpg	.jpg	417291	7/9/2003 11:39:35..
012.jpg	.jpg	415785	7/9/2003 11:39:33..
018.jpg	.jpg	507654	7/9/2003 11:39:33..
019.jpg	.jpg	81736	7/9/2003 11:28:46..
02.jpg	.jpg	49150	7/9/2003 11:39:33..
021.jpg	.jpg	446328	7/9/2003 11:39:33..
022.jpg	.jpg	411843	7/9/2003 11:39:33..
024.jpg	.jpg	120337	7/9/2003 11:29:02..
026.jpg	.jpg	99417	7/9/2003 11:29:02..

Figure 4.3.4 Five basic views of a region: list, large icon, small icon, details, and thumbnails (clockwise from the top-left region)

Besides the five basic views, three additional views (aggregation, intersection, and background) were implemented to provide a quick overview of the model as well as to represent the interrelationships of the regions. In the aggregation view, the region shows the number of personal media items that it contains. This information can be encoded as a color or a size of the region to assist quick perception (**Figure 4.3.5** (a)). The background view enables users to easily remember the implication of the region through a background image. It also helps users perceive the internal hierarchy of a region through the overview of the child regions (**Figure 4.3.5** (b)). On the other hand, the intersection view was designed to represent the amount of personal media items shared among the regions. More detailed descriptions about the intersection view appear in the following "Meaning Extraction" section.

The overall design of multiple views for *Semantic Regions* was based on the well-known data visualization strategy, "Overview first, zoom-in, then filter, and detail-on-demand" [13][59]. Users can overview the model and distribution of personal media data first, and then see more details by maximizing the region. The design was also based on the theory of query preview [27] or dynamic aggregation [15][25][66].



(a) Aggregation view with size and color encoding

(b) Background view with background image and overview of child regions

Figure 4.3.5 Advanced views of *Semantic Regions*

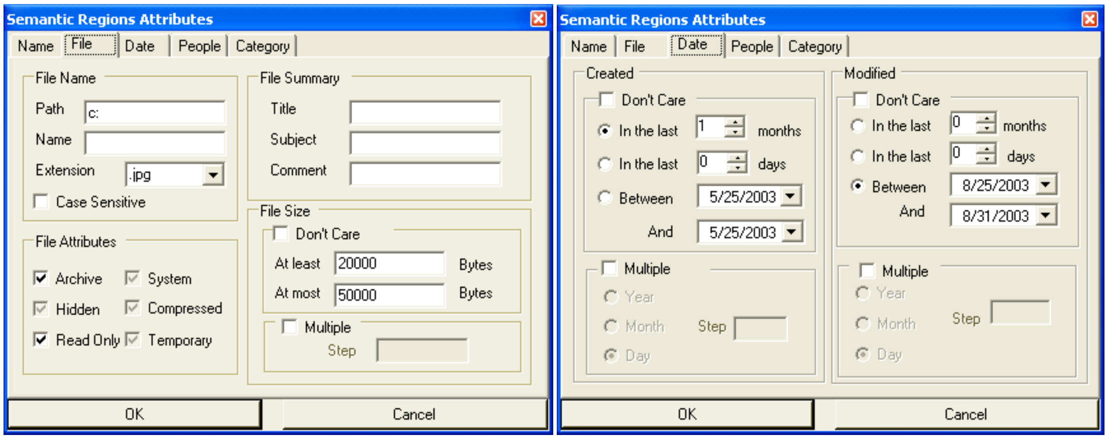
Users can draw lines among the *Semantic Regions* to illustrate the interrelationship among the regions more clearly. Users can group/ungroup the *Semantic Regions* to create/remove a hierarchy among the regions. MediaFinder provides several align and format functions to facilitate the construction of users' mental models. The functions include: making the selected regions have the same width or height, aligning the regions vertically(top, bottom, and middle) or horizontally (left, right and center), or adjusting the horizontal or vertical spacing among the regions.

4.3.2 Semantic Specification

Each *Semantic Region* ought to have semantics to enclose the personal media items. The semantics of a region consist of the file attributes of MS Windows file system(e.g. file name, path, date created/modified, size, type, etc) and the user defined high-level attributes (e.g. location, rating, people, season, event, etc.). These two different types of attributes are joined conjunctively to form the semantics of a region.

Left-clicking the mouse on a region pops up a menu. From this menu, users can select the “semantics menu”, which brings up the Semantic Regions Attribute dialog box. The Semantic Regions Attributes dialog box is composed of five tabs: name, file, date, people, and category. The name tab is used for specifying the title of the region. In the file and date tabs, users can make use of a variety of file attributes of the Windows file system to specify the semantics for a region (**Figure 4.3.6**). Once the attribute values are specified, all the personal media items that do not satisfy the criteria will be filtered out dynamically from the region. The “Multiple” check boxes located at the bottom of dialog box are used for specifying the semantics for multiple *Semantic Regions* (bulk semantic specification). This function can reduce users’ difficulties in specifying identical or continuously increasing/decreasing attribute values for multiple regions.

On the other hand, users can define their own category attributes and use them in specifying the semantics for a region. They can even define multiple categories and create a hierarchy for each category. For example, users can define both a location category that has a three-level (country-state-city) hierarchy and a season category that has just one-level hierarchy to specify the user-defined semantics for a region.


Semantic Regions Attributes

Name | File | Date | People | Category

File Name
Path c:
Name
Extension .jpg
□ Case Sensitive

File Summary
Title
Subject
Comment

File Attributes
☑ Archive ☑ System
☑ Hidden ☑ Compressed
☑ Read Only ☑ Temporary

File Size
□ Don't Care
At least 20000 Bytes
At most 50000 Bytes
□ Multiple
Step

OK Cancel

Semantic Regions Attributes

Name | File | Date | People | Category

Created
□ Don't Care
● In the last 1 months
○ In the last 0 days
○ Between 5/25/2003
And 5/25/2003
□ Multiple
○ Year
○ Month
● Day
Step

Modified
□ Don't Care
○ In the last 0 months
○ In the last 0 days
● Between 8/25/2003
And 8/31/2003
□ Multiple
○ Year
○ Month
● Day
Step

OK Cancel

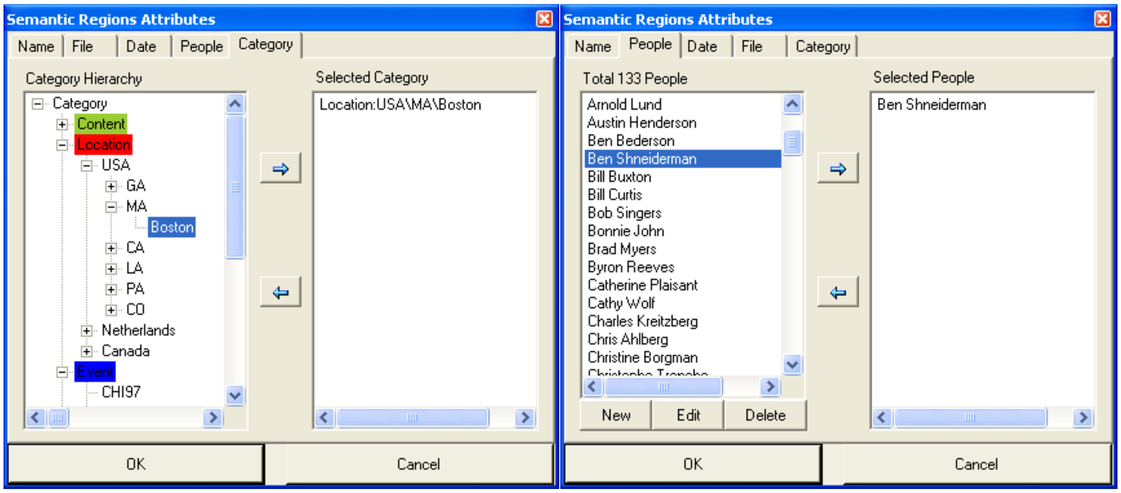
(a) In the file tab, users can specify several basic file properties such as name, summary, size, and attributes.
and date created

(b) In the date tab, users can specify two major file date properties for a region: date modified

Figure 4.3.6 In the Semantic Regions Attributes dialog box, the file tab and the date tab are used for specifying the system-oriented file attributes for a region

In the Semantic Regions Attributes dialog box, either the category tab or the people tab is used for specifying the user-defined attributes for a region. In the category tab, a user-defined attribute is specified by selecting a node from the category tree (left window) and adding (pressing the right arrow button) it to the selected category listbox (right window) (**Figure 4.3.7**(a)). The selected node is converted to a string just like a full path name of a folder in the windows file system. It starts with the category name that is followed by a colon and the series of ancestor node names separated by a back-slash character (e.g. location:USA\MD\College park). The specified categories can be removed from the selected category listbox by pressing the left arrow button. Users can create, modify, or remove a category node by using the "new", "edit", and "delete" buttons located at the bottom of the category hierarchy window.

Although the "People" category is one of many user-defined categories, MediaFinder was designed to have a separate "People" tab in the Semantic Regions Attributes dialog box because the names of people are frequently used as key metadata in personal media data. As in the category tab, users can create, modify, and remove a name from the People listbox through "new", "edit", and "delete" buttons in the people tab. People attributes are specified for a region by selecting a person and adding it to the selected people listbox. The selected people can be removed from the selected people listbox by pressing the left arrow button.



Selected Category
Location:USA\MA\Boston

Selected People
Ben Shneiderman

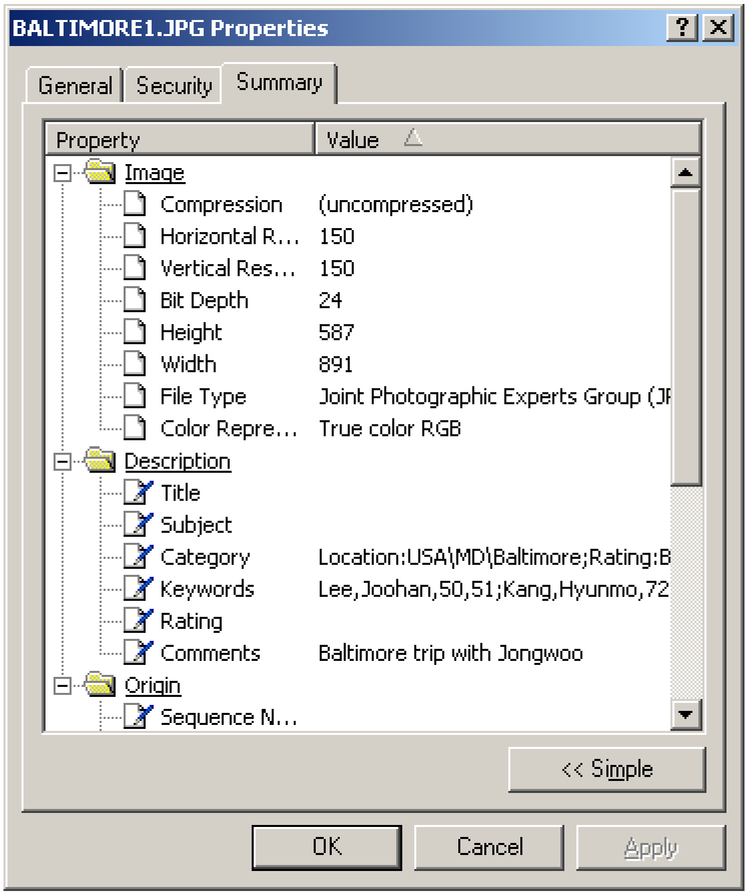
(a) Specification of category attributes (b) Specification of people attribute

Figure 4.3.7 Specifying the user-defined attributes for a region in the category and the people tab of the Semantic Regions Attribute dialog box.

All the categories and people attributes specified for a region are combined conjunctively. In other words, only the personal media items that have all of the specified attributes can be contained in the region. For example, if MediaFinder is bound to personal digital photos and a region in MediaFinder is specified with the category attribute “Location:USA\MD\College Park” and the people attribute “Hyunmo Kang”, then only the photos of Hyunmo Kang taken in College Park, Maryland will be contained in the region.

Personal media items can be annotated with user-defined attribute values such as category and people. As mentioned above, a user-defined attribute value is a string that starts with the name of a category and a colon, and is followed by a series of the names of sub-categories separated by a back-slash character. MediaFinder makes use of Microsoft Windows NTFS (Windows NT File system) file summary property to keep the user-defined attribute values of personal media items because the current file systems do not support a good way to combine the user-defined attributes with the personal media items. MS Windows NTFS provides the file summary properties so that users can read and write a description of a file and these properties are then used to store and retrieve the user-defined attribute values such as category and people in MediaFinder (**Figure 4.3.8**).

When personal media items are bound with MediaFinder through *fling-and-flock*, MediaFinder collects all the user-defined attribute values stored in NTFS file summary properties of the personal media items and builds the hierarchies of category and people attributes to represent in the Semantic Regions Attributes dialog box. These pre-constructed hierarchies of the user-defined attributes are used for specifying the semantics for a region.



Property	Value
Image	
Compression	(uncompressed)
Horizontal R...	150
Vertical Res...	150
Bit Depth	24
Height	587
Width	891
File Type	Joint Photographic Experts Group (JF
Color Repre...	True color RGB
Description	
Title	
Subject	
Category	Location:USA\MD\Baltimore;Rating:B
Keywords	Lee,Joohan,50,51;Kang,Hyunmo,72
Rating	
Comments	Baltimore trip with Jongwoo
Origin	
Sequence N...	

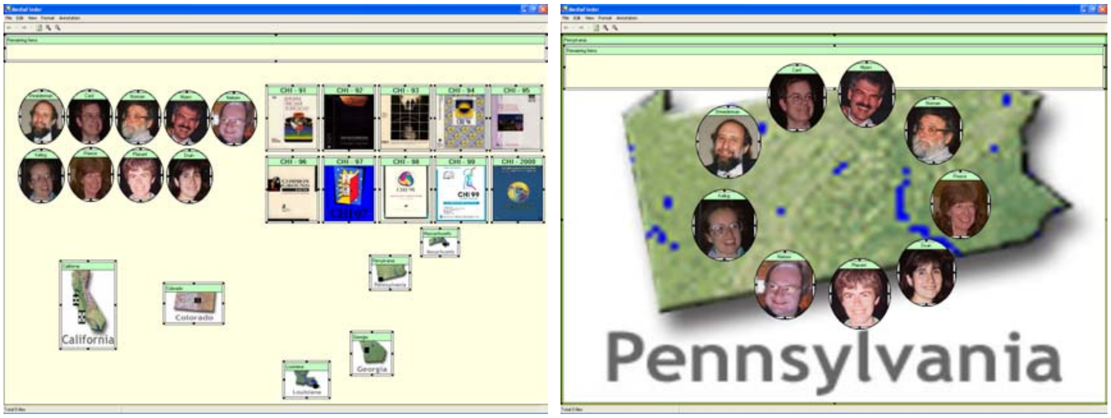
Figure 4.3.8 The File summary properties in Microsoft Windows NT File System. File descriptions are used to store and retrieve the user-defined attributes such as category, people, and comments in MediaFinder.

4.3.3 Model Management

The layout of the regions as well as the semantics of each region can be saved as a file for later use. Users can apply various different mental models to the same personal media data set by opening the saved files. Also, users can import and combine multiple heterogeneous models together in MediaFinder to observe the interrelationships among the models.

The models can be combined horizontally or vertically (**Figure 4.3.9**). Users can combine the multiple models on the same level so that all the models are applied to the same personal media items simultaneously. By combining the models horizontally, users can easily recognize the interrelationships among the models and formulate a query across the different models. On the other hand, users can import a model inside a region so that the imported model is applied only to the personal media items that are contained in the parent region. By combining the models vertically, users can dynamically regroup the personal media items in a region based on a variety of models.

The MediaFinder prototype was designed to have a four-level hierarchy overall: Model - Semantic Region - Personal Media Item - Semantics (Attributes). In other words, a model is composed of several *Semantic Regions* and each region contains personal media items that are specified with the semantics (both file system attributes and user-defined attributes).



(a) Three models combined horizontally (b) People model inside a location region

Figure 4.3.9 Multiple models can be combined horizontally or vertically

4.4 Using Semantic Regions in MediaFinder

Binding a *Semantic Regions* model with the personal media items enables users to use the MediaFinder as a tool for personal media management. The personal media management consists of several tasks: organization, meaning extraction, search, navigation, indexing, and distribution. The following sections describe how

MediaFinder was designed and implemented to deal with each personal media management task.

4.4.1 Data Binding (the *fling-and-flock* Metaphor)

Once the *Semantic Regions* are created, they are ready to be applied to personal media items. When users drag and drop the personal media items from the Windows file explorer onto the model, the items are automatically distributed and placed in the appropriate regions based on the semantics defined in each region. This metaphor is called "fling-and-flock". That means, users fling the personal media items, and the items flock to the regions.

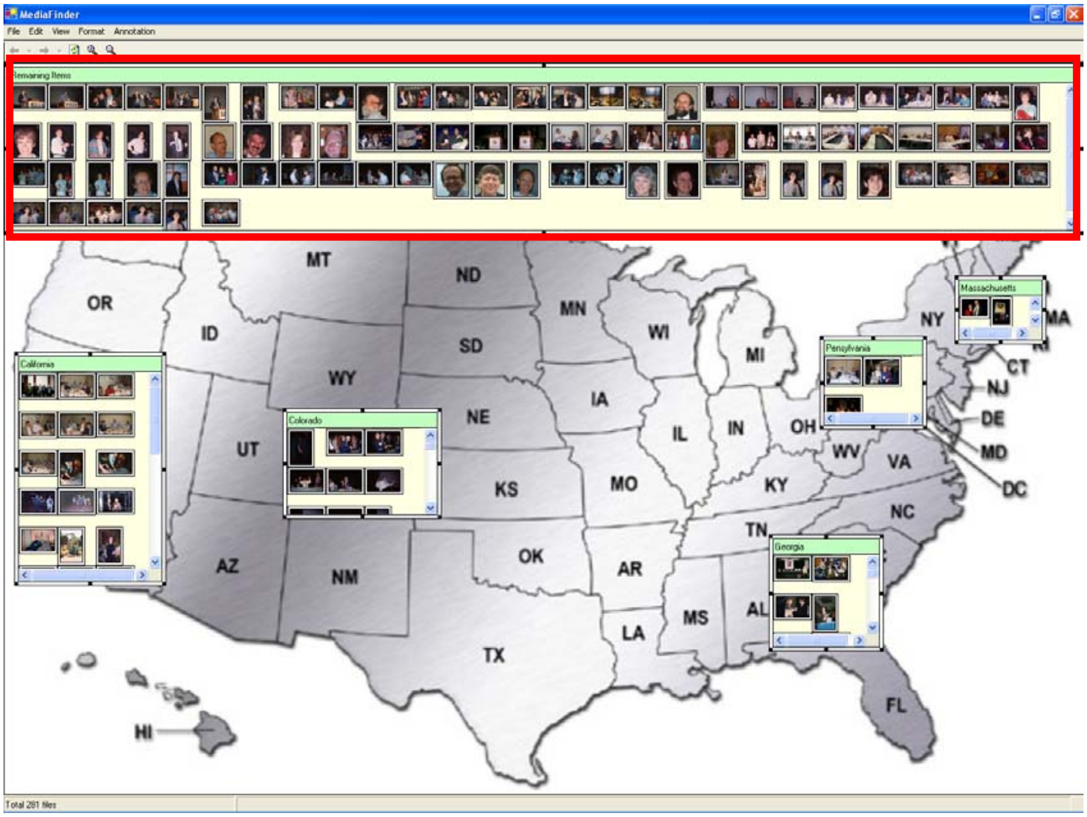
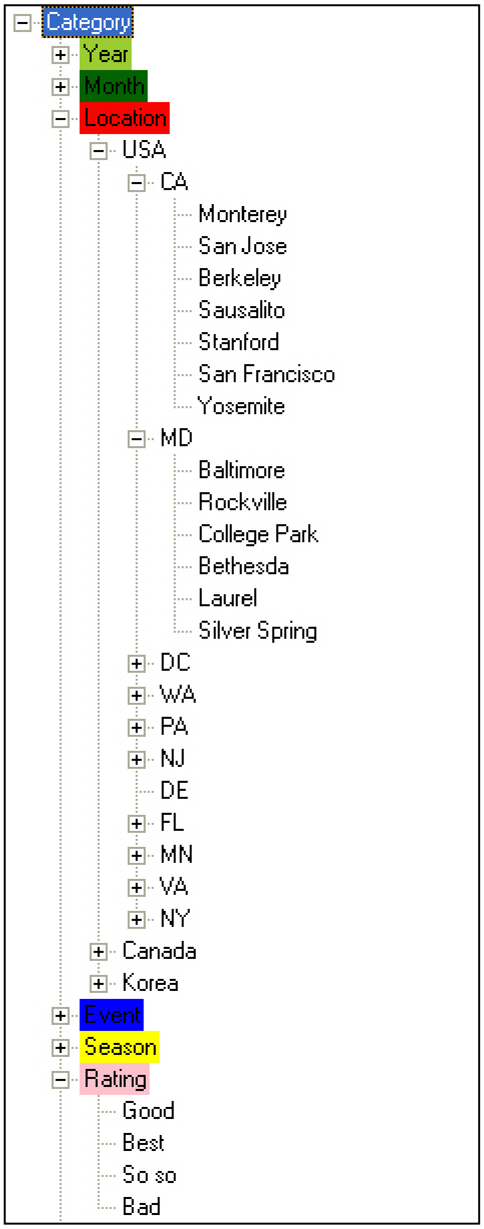


Figure 4.4.1 The photos that do not satisfy the semantics of five state regions are collected in the "Remaining Items" region (red rectangle) located at the top of MediaFinder main window.

If the personal media items do not satisfy any of the semantics of the regions, they are collected into the “Remaining Items” region (**Figure 4.4.1**). The “Remaining Items” region is a specially designed *Semantic Region.* Unlike other *Semantic Regions,* it is not allowed to have any semantics so that it can contain all the personal media items that do not satisfy any of the criteria of the region in the same level. Likewise, each region was designed to have a child “Remaining Items” region to contain all the personal media items that do not satisfy any of the criteria of the child regions but which are contained in the parent region.

The relationship between a *Semantic Region* and a personal media item is similar to the relationship between a folder and a file. A *Semantic Region* can include the child regions as well as the personal media items just as a folder can contain the subfolders as well as the files. However, there exists a major difference in that a personal media item can appear in the multiple regions as long as it satisfies their semantics but a file cannot appear in multiple folders.

During the *fling-and-flock* process, MediaFinder collects all the attribute values (both the file system attributes and the user-defined attributes) of the bound personal media items and keeps them in the Semantic Regions Attributes dialog box. The collected attribute values are used for specifying the semantics for a region or for giving an overview of what kinds of metadata are annotated to the personal media items just like query preview [27]. The collected user-defined attributes are translated to form a semantic hierarchy as shown on the right. **Figure 4.4.1** shows that a total of 6 user-defined categories were collected from the personal media items and each category had its own hierarchy. For example, the location category has a three level



- Category
  - Year
  - Month
  - Location
    - USA
      - CA
        - Monterey
        - San Jose
        - Berkeley
        - Sausalito
        - Stanford
        - San Francisco
        - Yosemite
      - MD
        - Baltimore
        - Rockville
        - College Park
        - Bethesda
        - Laurel
        - Silver Spring
      - DC
      - WA
      - PA
      - NJ
      - DE
      - FL
      - MN
      - VA
      - NY
    - Canada
    - Korea
  - Event
  - Season
  - Rating
    - Good
    - Best
    - So so
    - Bad

Figure 4.4.1 User-defined attributes collected by MediaFinder through fling-and-flock

hierarchy (country-state-city), but the Rating category has just one level hierarchy to represent the rating of the personal media items.

There are two different ways to bind the personal media items to MediaFinder: static binding and dynamic binding. In static binding, users can specify a folder that they want to explore and then all the files in the specified folder and the subfolders are bound to MediaFinder. Users can change the pre-bound folder anytime if necessary. The static data binding is useful when users want to explore the computer file systems with Mediafinder. On the other hand, dynamic binding enables users to select and bind only the personal media items that they are interested in to MediaFinder through the *fling-and-flock* metaphor. The dragged personal media items are bound to MediaFinder and automatically distributed and placed in the appropriate regions based on the semantics of each region. This metaphor is called “fling-and-flock”. Users can change the bound personal media items by dragging another set of personal media items onto MediaFinder. Once the personal media items are bound to MediaFinder through either static or dynamic data binding, any change that occurs in *Semantic Regions* is updated dynamically as in dynamic queries[3][2][22][75].

With the data binding mechanisms designed in MediaFinder, users can easily apply different mental models to the same personal media items. The personal media items bound to MediaFinder are dynamically grouped and distributed over the *Semantic Regions* whenever users open a new model or combine multiple models by importing them in MediaFinder.

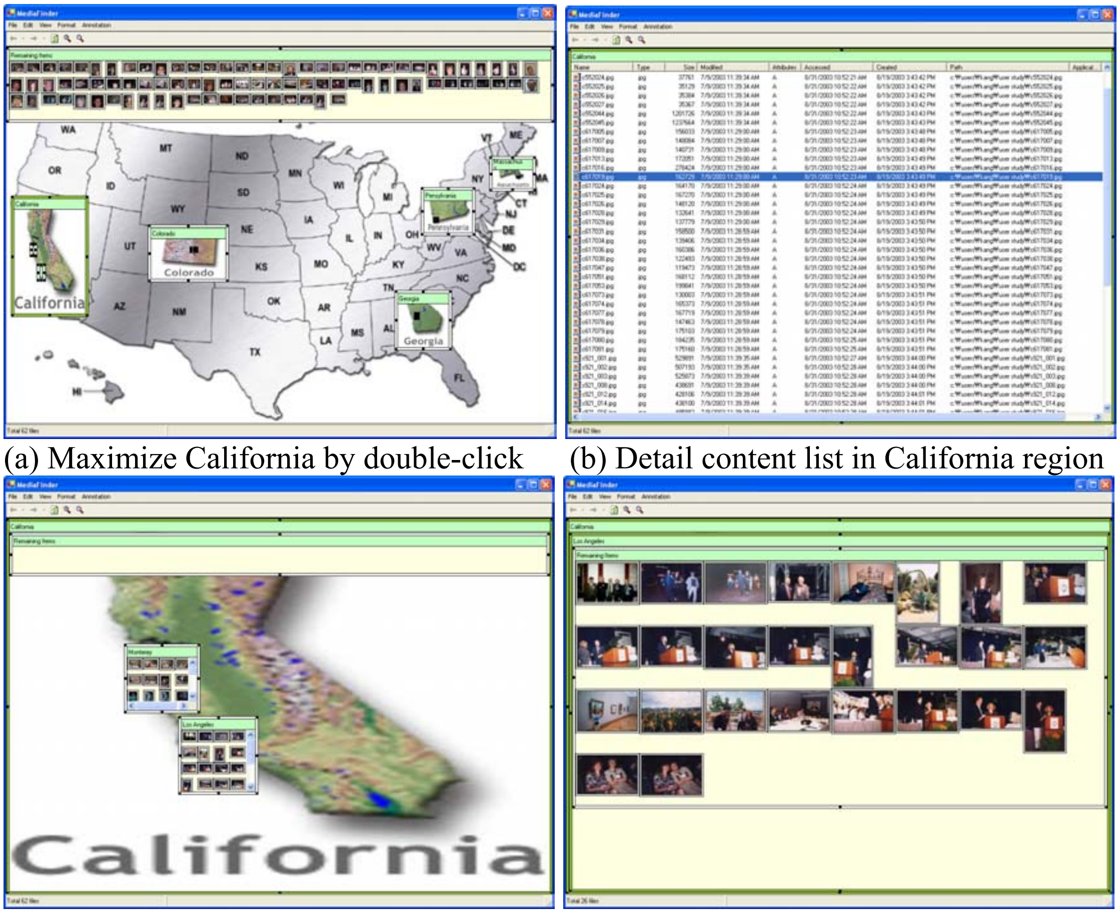
4.4.2 Flexible Organization

Each personal media item can be contained in multiple *Semantic Regions* as long as it satisfies the criteria of the regions. The semantics of the regions do not need to always be mutually exclusive. In other words, the intersection of any two regions in MediaFinder does not have to be empty. This design is based on the fact that users' mental models for categorizing personal media are not always mutually exclusive unlike the file system structure (i.e. In the file-folder hierarchy system, a personal media appears in only one folder at a time). For example, suppose that users want to organize their digital photos based on people who appeared in the photos. Then, the photos in which more than one person appeared should be contained in multiple collections.

This flexible organization of personal media items also makes it possible for users to combine different mental models and observe the interrelationship among them. The multiple inclusion of a personal media item across the regions often depicts both the intra-relationships and the inter-relationships of the different models. The interactive visualization of the interrelationship (Region Brushing) of the combined models will be explored in detail in the Meaning Extraction section.

4.4.3 Navigation

Semantic Regions can be maximized to show the detail information of the contained personal media items or to represent the hierarchy of the child regions (**Figure 4.4.2**). **Figure 4.4.2**(a) shows the USA map model that contains five state regions. A region is maximized to fill the parent region with animation when it is double-clicked. **Figure 4.4.2**(b) shows the maximized California region with a detail view to display the list of the files contained in the region.



(a) Maximize California by double-click (b) Detail content list in California region

(c) Two city child regions in California (d) Maximize a city region recursively

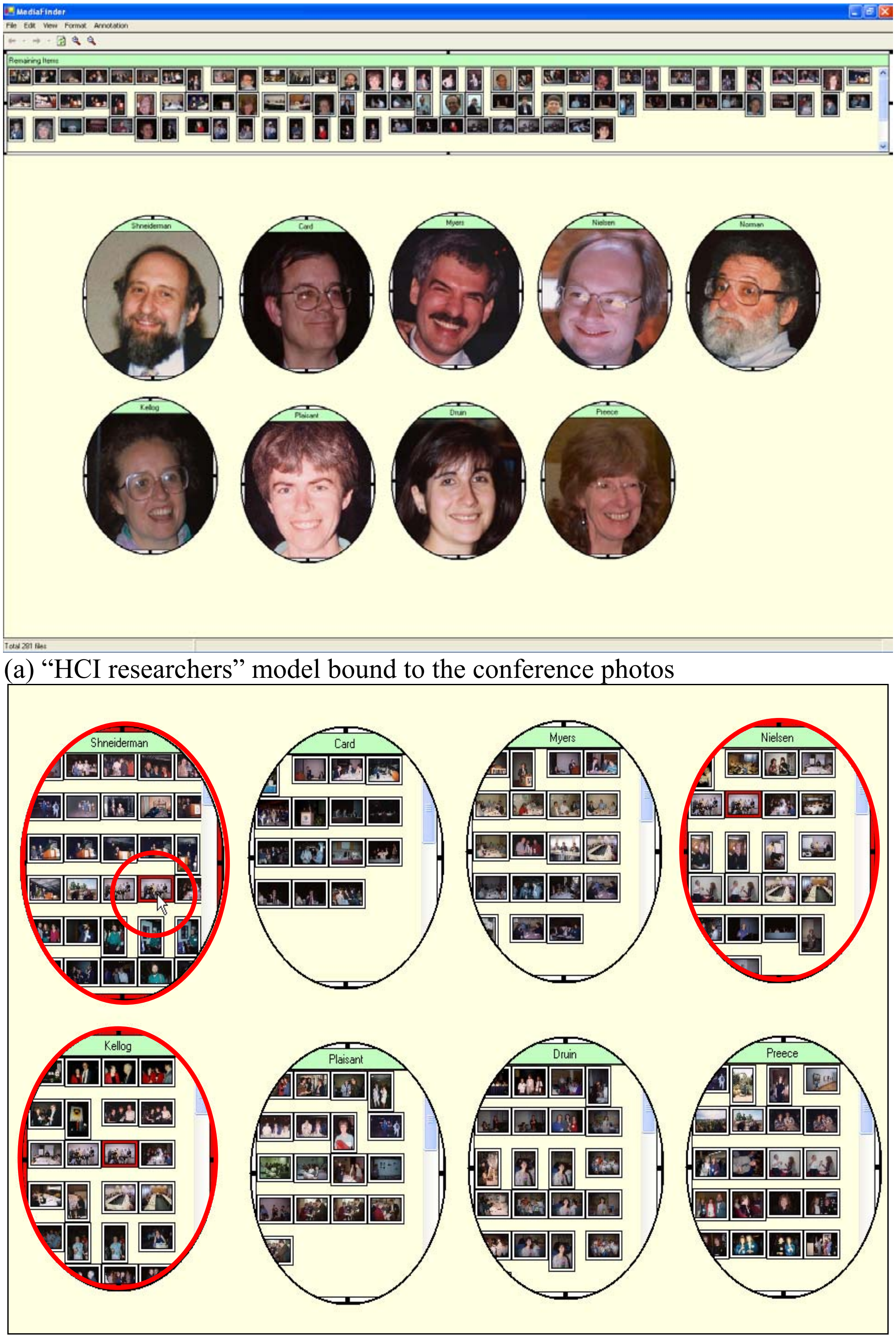
Figure 4.4.2 The maximized region shows more details about the contained personal media items or represents the hierarchy of child regions.

The background view shows the child regions located inside the California region. The personal media items contained in the California region are redistributed and placed in the two city (Los Angeles and Monterey) regions (**Figure 4.4.2**(c)). In **Figure 4.4.2**(d), the Los Angeles city region is maximized recursively and shows the contained items with a thumbnail view. The titles of the entire maximized regions are displayed contiguously to represent the current level in the hierarchy (**Figure 4.4.2**(d)). The maximized region shrinks back to its original size when it is double-clicked again. All of the maximizing and minimizing processes are animated to reduce the users' confusion that might be caused by the change of region hierarchy.

4.4.4 Meaning Extraction

MediaFinder provides an interactive visualization technique called "Region Brushing" to visualize the relationships among the *Semantic Regions*. Region Brushing was based on the idea of flexible organization of personal media items. Basically, Region Brushing is used to highlight the same personal media items contained in multiple regions simultaneously and is often used for extracting meanings of the personal media items bound to MediaFinder.

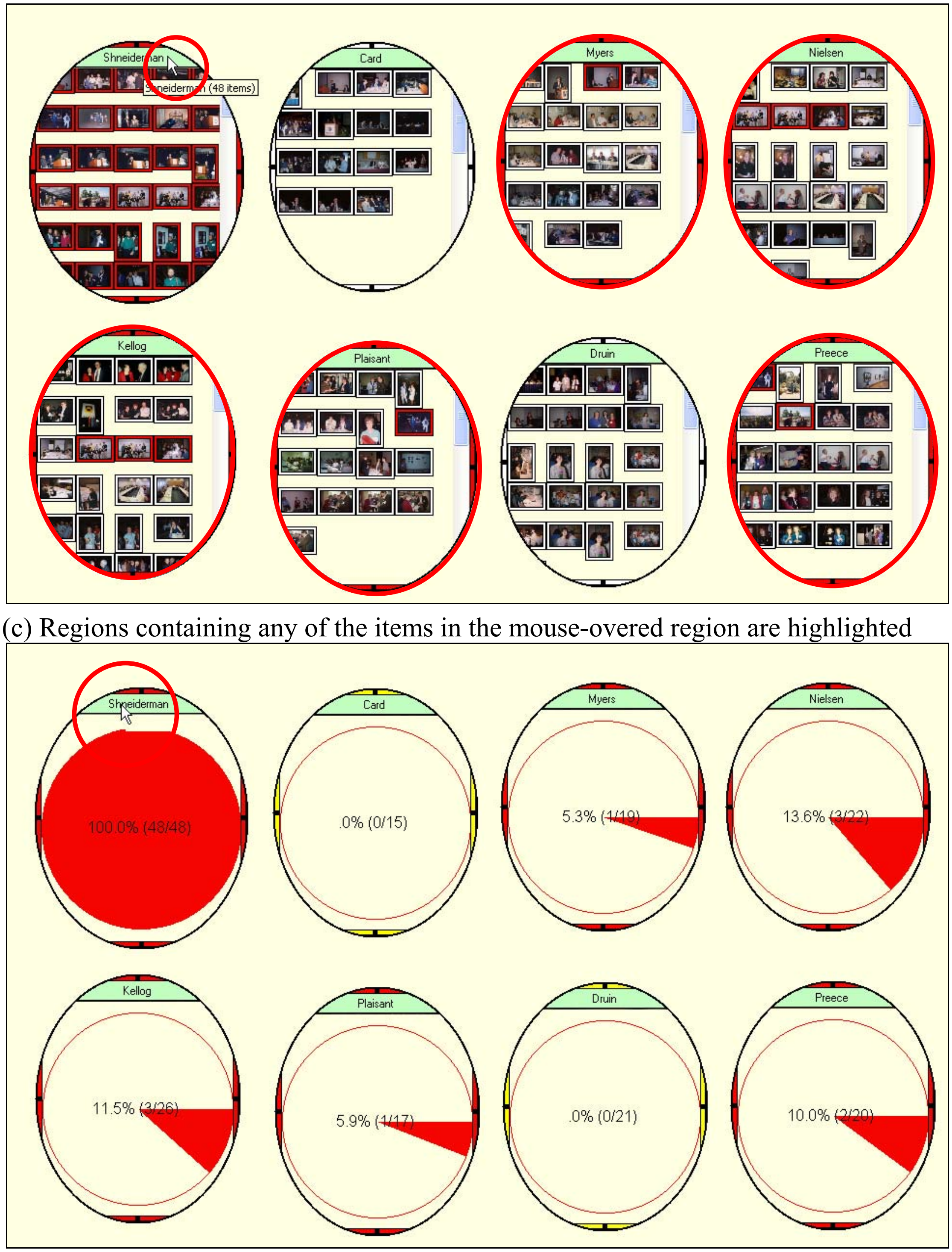
Since a personal media item can be contained in multiple *Semantic Regions*, the existence as well as the amount of the shared personal media items among the regions well represents their relationships. MediaFinder dynamically visualizes this information through Region Brushing. **Figure 4.4.3**(a) shows a "HCI Researchers" model in which each region represents a person. The conference photos were bound to this model through *fling-and-flock* so that each photo could be placed in multiple people regions based on its annotations.



(a) “HCI researchers” model bound to the conference photos

(b) The selected item and all the containing regions are highlighted red

Figure 4.4.4 Region Brushing: an interaction technique to visualize the shared items among *Semantic Regions* (Continued)



(c) Regions containing any of the items in the mouse-overed region are highlighted

(d) Intersection view to represent the amount of shared items among the regions

Figure 4.4.3 Region Brushing: an interaction technique to visualize the shared items among *Semantic Regions*

Region Brushing works as follows. When the mouse is over an item in a region, borders of the regions that contain the item are highlighted red. The selected item in the highlighted regions is also highlighted to be easily identified (**Figure 4.4.3**(b)). In addition, when the mouse is over the title of a region, all the regions that contain any of the items in this region are highlighted red while highlighting the containing items (**Figure 4.4.3**(c)).

In addition, *Semantic Regions* was designed to provide an "Intersection view" to visualize how many personal media items in the selected region are shared with the other regions (**Figure 4.4.3**(d)). With this interactive visualization technique, users can quickly recognize the relationships among regions and thus answer questions like, "Who appeared in pictures with Ben Shneiderman?", "How many pictures were taken with Ben Shneiderman?", or "What percentage of the pictures of a person were taken with Ben Shneiderman?".

The Region Brushing technique becomes even more powerful when multiple models are combined horizontally to represent interrelationships among the models in MediaFinder. In **Figure 4.4.4**, three models, HCI researcher, US map, and CHI conference calendar are combined horizontally (in the same level) and bound to the conference photos through the *fling-and-flock* metaphor.

Since a photo can be contained in multiple regions across the different models, many interesting questions concerning the interrelationships of multiple models can be answered through Region Brushing. Such questions might be "Find the name of the conferences that all nine people participated in and where was the conference held?", "What was the name of the conference held in Atlanta, Georgia, and who did not appear in the photos taken at this conference?", "Find the conferences that Ben Shneiderman did not appear in the photos and where was the conference held?", "Find the conferences that were not held in the US", and so on.

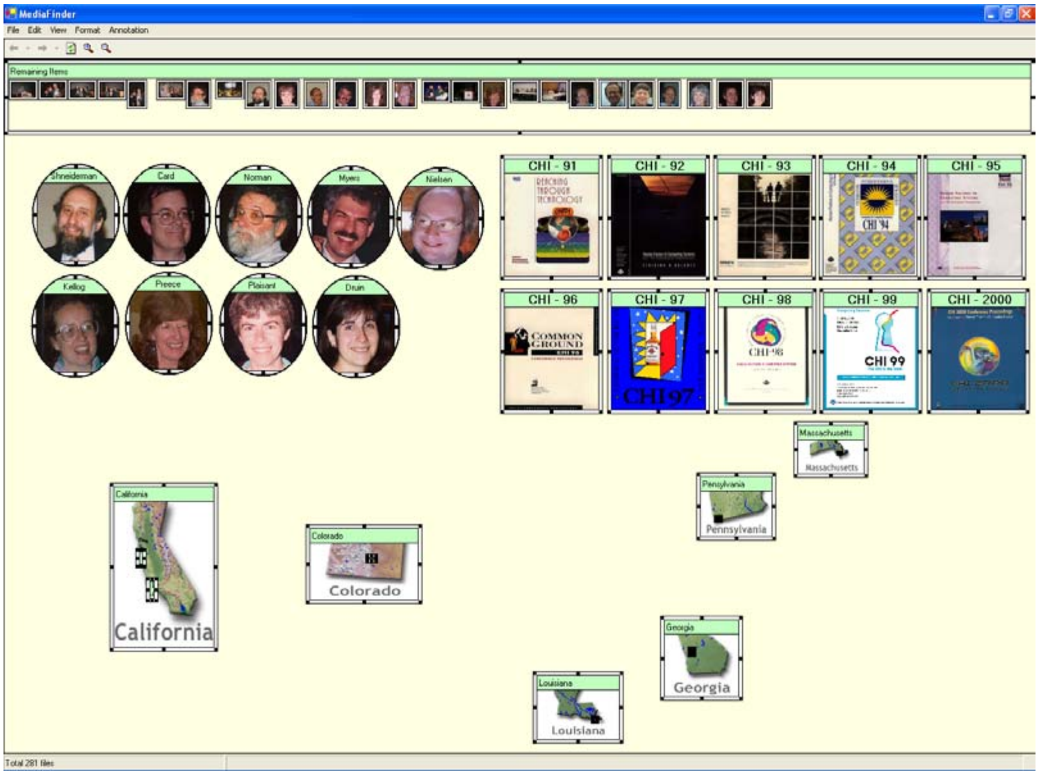
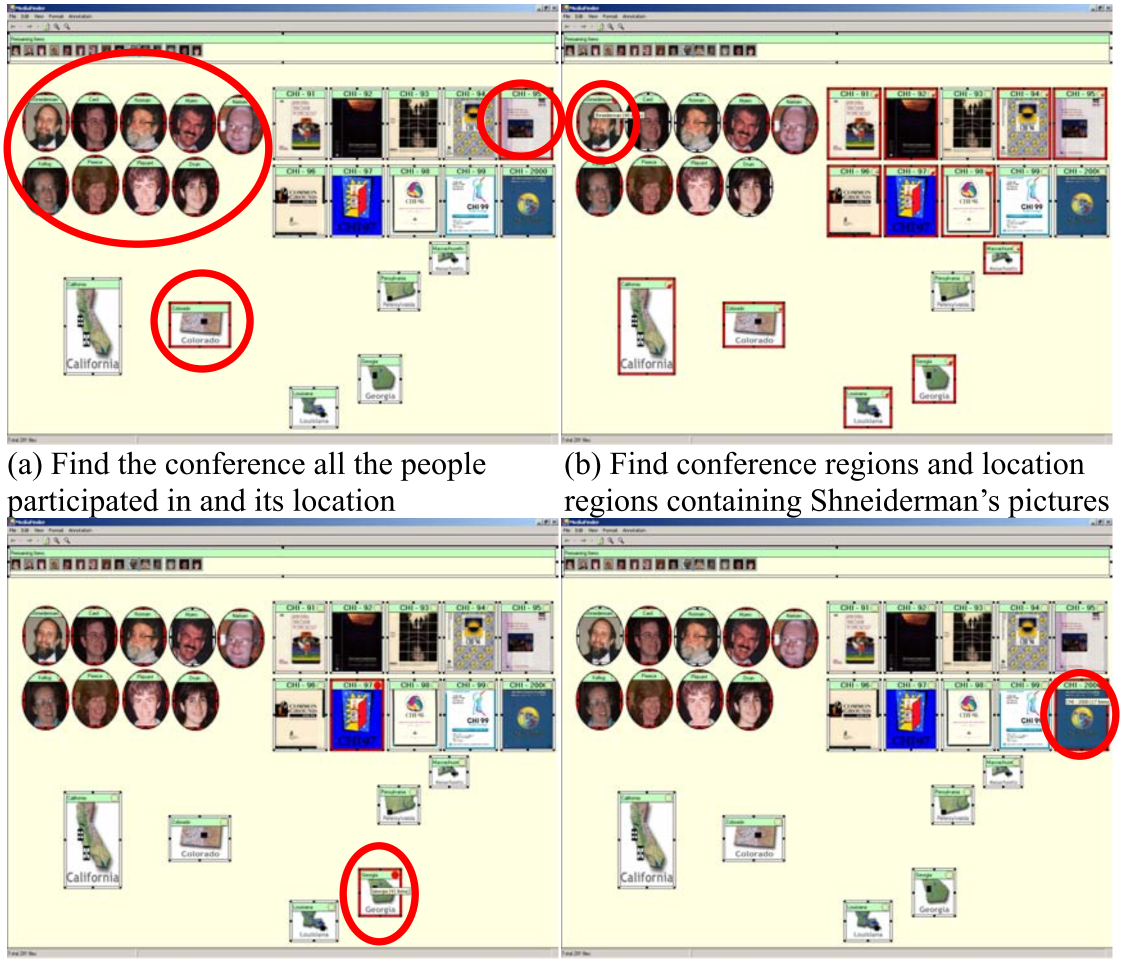


Figure 4.4.4 Three models are combined horizontally to extract meanings about the interrelationships among the different models through Region Brushing.

The first question can be answered by placing the mouse cursor over the conference regions one at a time. If all the people regions are highlighted red when the mouse is over a certain conference region, it means that every person region contains at least one picture that the mouse-overed conference region contains (**Figure 4.4.5** (a)). For the second question, users can easily find the answer by placing the mouse over the Georgia state region and observing what conference region is highlighted and which people regions are not highlighted (**Figure 4.4.5** (b)). The third problem can easily be solved by placing the mouse over the Ben Shneiderman region and observing which conference regions as well as which state regions are not highlighted (**Figure**

4.4.5 (c)). The fourth question can be answered by placing the mouse over the conference regions one at a time and observing which conference does not cause any of the state regions to be highlighted (**Figure 4.4.5** (d)).



(a) Find the conference all the people participated in and its location

(b) Find conference regions and location regions containing Shneiderman's pictures

(c) Find the conference held in Georgia and the people who participated in it

(d) Find the conference not held in the US

Figure 4.4.5 Extracting meanings concerning the interrelationships of the multiple models through the Region Brushing technique

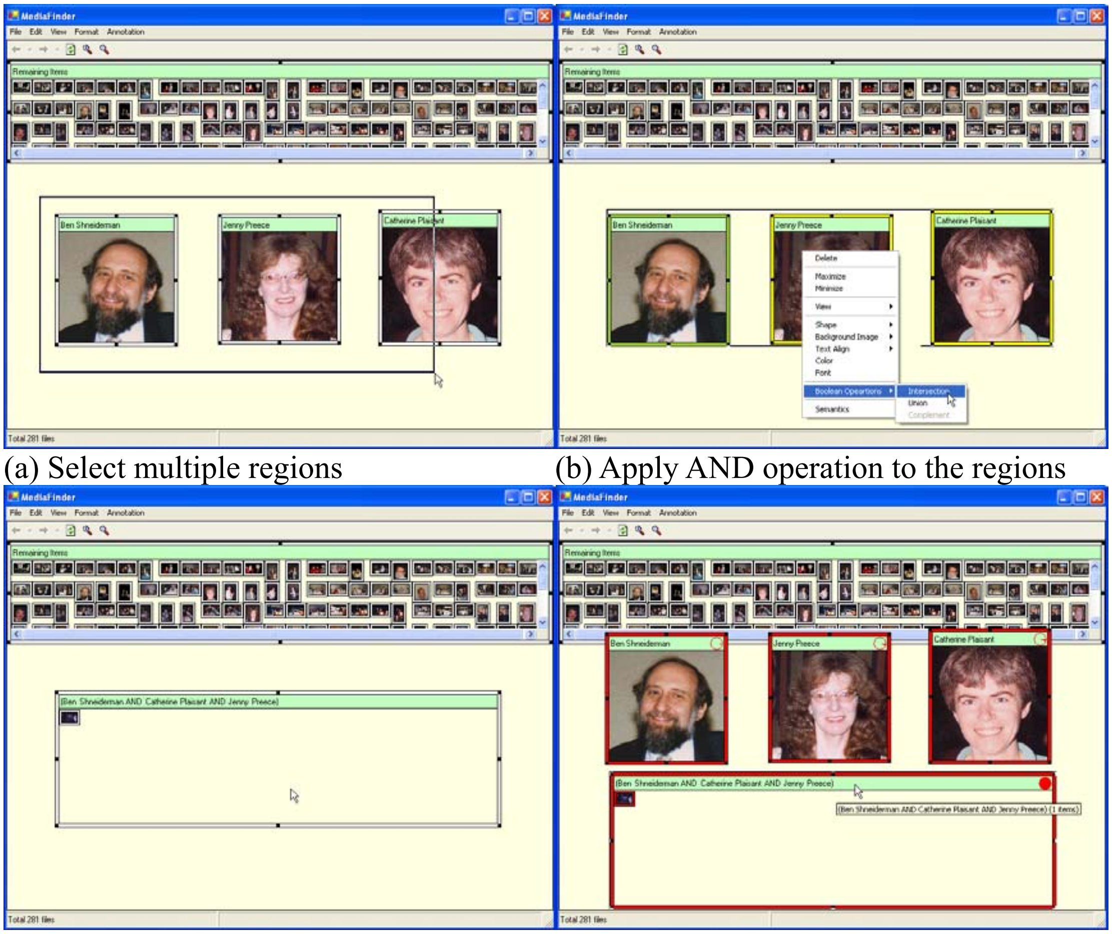
From the above examples, it is obvious that users can extract diverse meanings concerning the interrelationships of the combined models by changing the view points of Region Brushing. In other words, users can discover different meanings about the interrelationships of the combined models depending on which model gets the major focus.

4.4.5 Search

MediaFinder supports Boolean operations (AND, OR, NOT) of the regions to facilitate a search. The design is based on the fact that the relationship between *Semantic Regions* and personal media items is the same as the relationship between sets and elements. While users can specify the semantics for a region by combining file system attributes and user-defined attributes conjunctively, they are not allowed to define the semantics that stand for disjunctive or complement combination of the attributes. As a matter of fact, this design was based on the belief that users tend to specify the semantics for a region as simply as possible in order to minimize the effort required to remember the implication of a region. This assumption is supported by the user study about the construction of *Semantic Regions* in the following chapter. Therefore, the basic design principle of MediaFinder was to make the semantics of a region as simple as possible and allow users to specify more complicated semantics, if necessary, by applying Boolean operations to the regions.

In MediaFinder, users can apply Boolean operations (AND, OR, and NOT) or set operations (Intersection, Union, and complement) to *Semantic Regions.* Boolean AND or OR operations are applied to multiple regions but the NOT operation is not allowed to be applied to more than one region since NOT is a unary operator. When an AND operation is applied to the selected regions, a new region is created over the regions to contain only the personal media items that are contained in every selected region. On the other hand, if the OR operation is applied to multiple regions, a new

OR region is created in order to contain the personal media items that are contained in any of the selected regions. If a NOT operation is applied to a region, a new region is created right over the selected region to contain all the personal media items that are not contained in the selected region but bound to MediaFinder. The Boolean operations can be recursively applied to the Boolean regions (the regions created by applying Boolean operations). With the Boolean operations, users can specify any logical combination of the semantics for a region.



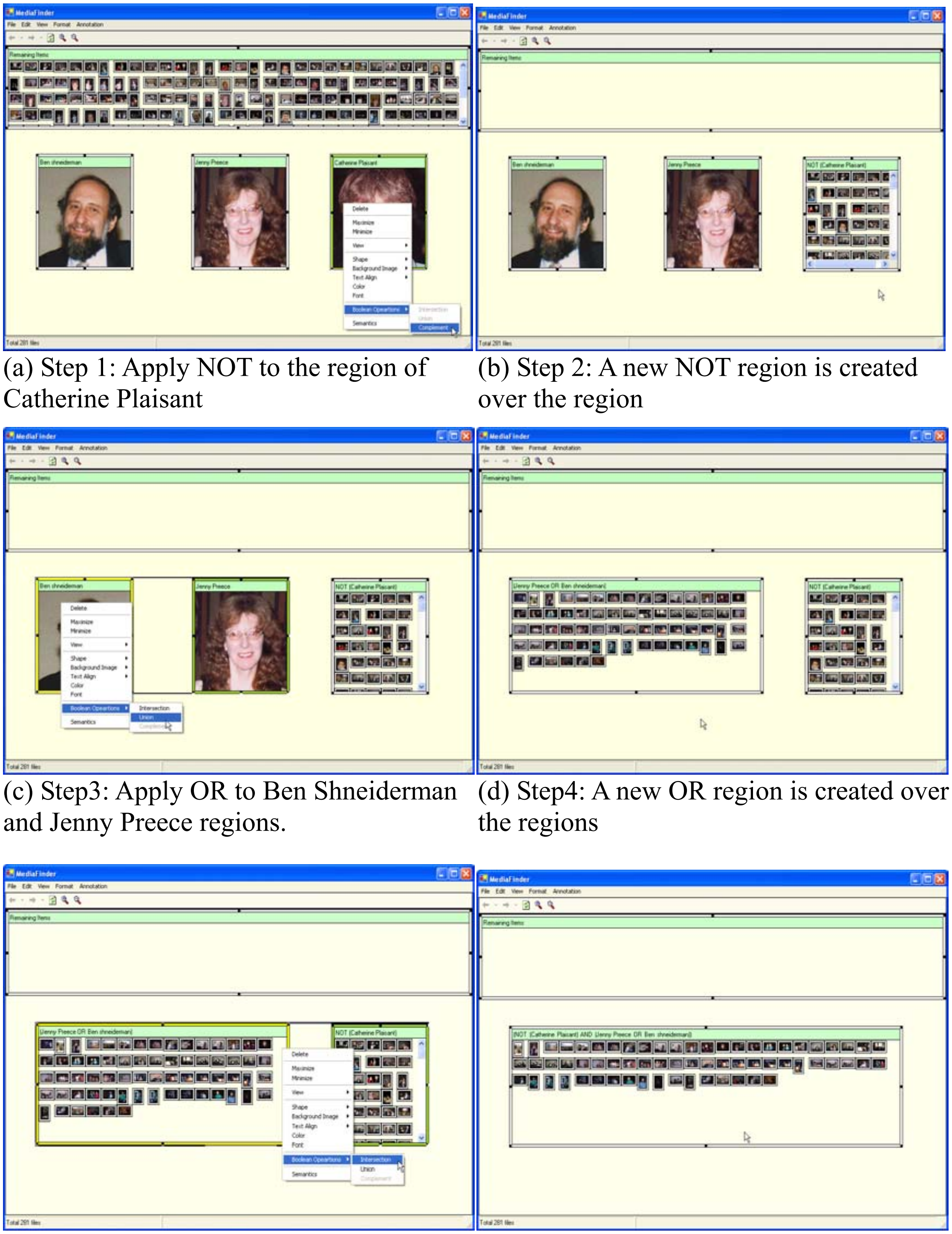
(a) Select multiple regions (b) Apply AND operation to the regions

(c) AND region is created over the regions (d) AND region makes regions highlighted

Figure 4.4.6 Applying the Boolean AND operation to the multiple regions

Figure 4.4.6 shows the process of applying an AND operation to the selected regions. First, users should select multiple regions that an AND operation will be applied to (**Figure 4.4.6**(a)) and then select the "Boolean Operation–Intersection" from the popup menus of *Semantic Regions* (**Figure 4.4.6**(b)). A new region is created over the regions as large as the bounding rectangle of the selected regions (**Figure 4.4.6**(c)). A newly created AND region contains only the personal media items that are contained in all three regions. In **Figure 4.4.6**(d), the AND region was moved from the original position to show that the item contained in the AND region is also contained in all three regions.

More complex Boolean queries can be achieved by repeatedly applying Boolean operations to the regions. Suppose an example, "Find all the photos in which either Ben Shneiderman or Jenny Preece appeared but not with Catherine Plaisant". A total of three Boolean operations are needed to achieve this goal. **Figure 4.4.7** describes all the steps to get an answer for this problem. In **Figure 4.4.7**(a), the procedure starts with applying the NOT operation to the region of Catherine Plaisant by choosing a "Complement" menu from the popup menus. Then the NOT region is created to contain the photos that are bound to MediaFinder but not annotated with the name, Catherine Plaisant (**Figure 4.4.7** (b)). The third step is to select both the Ben Shneiderman region and the Jenny Preece region to apply an OR operation (**Figure 4.4.7** (c)). The OR region is created over the selected regions and it includes all the photos that are annotated with either Ben Shneiderman or Jenny Preece (**Figure 4.4.7** (d)). Finally, the AND operation is applied to the NOT and the OR regions (**Figure 4.4.7** (e)), and the AND region is created to contain the photos that only satisfy the Boolean query (NOT (Catherine Plaisant)) AND ((Ben Shneiderman) OR (Jenny Preece))).



(a) Step 1: Apply NOT to the region of Catherine Plaisant

(b) Step 2: A new NOT region is created over the region

(c) Step3: Apply OR to Ben Shneiderman and Jenny Preece regions.

(d) Step4: A new OR region is created over the regions

(e) Step5: Apply AND to two Boolean regions

(f) Step6: Final AND region constructed over the OR and the NOT regions

Figure 4.4.7 The steps for applying Boolean operations to get the Boolean region of (NOT (Catherine Plaisant)) AND ((Ben shneiderman) OR (Jenny Preece)))

Boolean regions do not have any user-specified semantics but they contain the information about what Boolean operation was applied (operation) and what regions were used as operands for this operation (operand regions). If there is any change in the contents of an operand region caused by modifying the semantics of the region or binding a new set of personal media items, all the Boolean regions that use this region as an operand would be updated together.

Recursively, the content change of the Boolean regions may also affect other Boolean regions that use the updated Boolean regions as their operand regions. Therefore, the Boolean regions are updated successively based on the relationships with their operand regions and thus the order of Boolean operations forms a hierarchy. Cycles are not allowed in the hierarchy of Boolean operations.

MediaFinder was designed to have a hierarchy mechanism for a Boolean operation priority to deal with this issue. All the user-specified *Semantic Regions* (also called a base region) are assigned a priority zero and a Boolean region is assigned a priority that is the maximum priority of its operand regions plus one. This rule can be converted to a mathematical formula as follows:

n: Number of user-specified regions

m: Number of Boolean regions

Region(i): User-specified *Semantic Region* **(*i = 1, …, n*)**

Region(j): Boolean Region **(*j = n+1, …, n+m*)**

Operand(Region(j), k): Operand region of Boolean region ***Region(j)***

(*n+1 ≤ j ≤ n+m, k=1, …,l(j), l(j)*): Number of operand regions of ***Region(j)*)**

Priority (Region(i))* = 0, if *1≤ i ≤ n

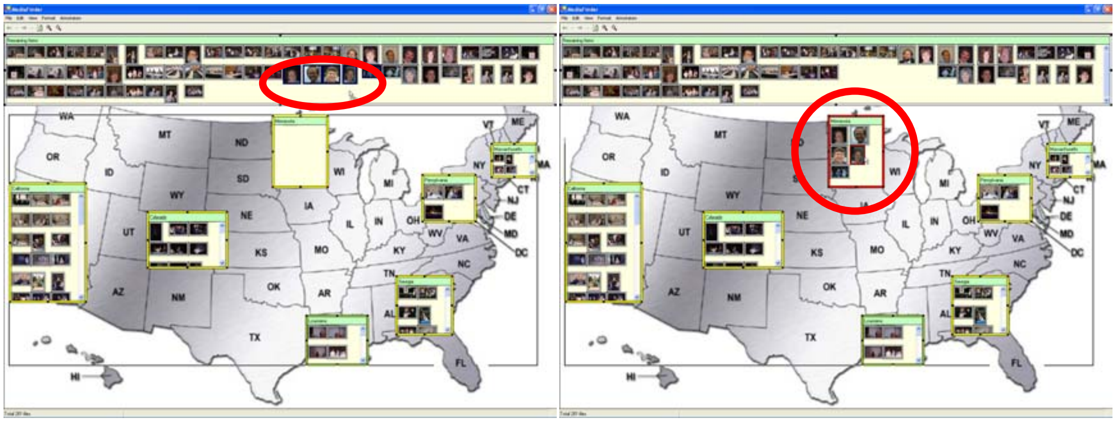
$$\begin{gathered} \mathit{Priority}\ (\mathit{Region}(j)) = \\ \mathit{Max}\ (\mathit{Priority}(\mathit{Operand}(\mathit{Region}(j),\ 1)),\ \ldots,\ \mathit{Priority}(\mathit{Operand}(\mathit{Region}(j),\ l(j)))) + 1 \\ ,\ \text{if}\ n+1 \le j \le n+m \end{gathered}$$

Whenever a new region is created in MedaFinder, a priority is assigned to it according to the above formula. And if there is any change in the contents (the contained personal media items) of a Region(j), Boolean regions are updated successively in increasing order from Priority(Region(j)) to the Maximun priority of Region(i) (i=1,…, n+m). With this priority system, Boolean regions can be updated dynamically without any conflict whenever there is any change in the base *Semantic Regions*. More details about the algorithms and data structures are explored in the following chapter.

4.4.6 Indexing (Annotation)

Semantic Regions can be used for indexing personal media items. If users select personal media items in one region and drag them to another region, the attribute values of the selected items are updated with those specified in the dragged region. If the semantics of the dragged region and the semantics of the original region are not mutually exclusive (In other words, if a personal media item can be contained in both of the regions simultaneously without any conflict of the semantics), the personal media items are duplicated and contained in both regions. On the other hand, if the semantics in the dragged region conflict with the semantics of the original region, the personal media items are removed from the original region and added to the dragged region.

In MediaFinder, the drag-and-drop metaphor is used for indexing personal media items just as it is used in direct annotation [61]. In **Figure 4.4.8** (a), five photo items in the remaining items region are selected and dragged onto the Minnesota region. If the items are dropped on the region, they are annotated with the location attribute value “location:USA\Minnesota” that is specified in the Minnesota region. Subsequently, the dragged items are removed from the remaining items region and added to the Minnesota region because they now satisfy the semantics of the region (**Figure 4.4.8** (b)).



(a) 5 photos are dragged from remaining items region onto the Minnesota region

(b) Dragged items are annotated with the semantics of the Minnesota region

Figure 4.4.8 Indexing the personal media items by dragging them onto the *Semantic Region*

4.4.7 Distribution

One major personal media management task is to distribute the personal media items either in private or in public. MediaFinder was designed to provide a few distribution features even though they were not the focus of this research.

The basic unit of distribution in MediaFinder is a region. Users can copy the personal media items in a region to the computer file system by dragging them to a folder in the Windows file explorer or through the copy-and-paste function. *Semantic*

Regions were also designed to be associated with the default email client so that the personal media items in a region can be sent to other users through email attachments. In addition, MediaFinder can generate web pages for each region to make the personal media items in the regions be downloadable through the web server. Each region can also be associated with a default ftp client to transfer the items through the file transfer protocol.

4.5 Summary

The MediaFinder user interface enables users to explore and manage personal media data by quickly constructing *Semantic Regions* that are appropriate for their specific tasks or mental models. Users first create regions on MediaFinder and then specify the semantics for each region. Personal media items are organized over the created regions through either static or dynamic data binding. In dynamic data binding, the personal media items are automatically distributed and placed in the appropriate regions based on the semantics defined in each region. This is done by dragging them onto the MediaFinder main panel. This metaphor is called *fling-and-flcok*.

In addition to the organization of personal media data, MediaFinder also enables a host of additional features that further amplifies its value. Multiple models can be combined together to represent the interrelationships between them. The Region Brushing technique is used for extracting the meaning from the combined model. Boolean operations can be applied to *Semantic Regions* to facilitate a search and navigation. *Semantic Regions* is also used for indexing and distribution of personal media data. Several enhancements to the MediaFinder interface will also be described in Chapter 6 based on results from user studies.

Chapter 5:

MediaFinder Implementation

MediaFinder was implemented in Visual Basic .Net, using Microsoft Visual Studio .Net 2003 for user-interface widgets. Initial versions of MediaFinder were built on top of the PhotoFinder to manage personal photos using Microsoft Visual Studio 6.0. *Semantic Regions*, personal media items, user-defined semantics, control panels and several dialog boxes were implemented as Microsoft COM/ActiveX components. After the first prototype of MediaFinder, the code was redesigned to replace Visual Basic 6.0 with Visual Basic .Net, a new object-oriented programming language intended to replace Visual Basic 6.0 to use the .Net framework.

As a research prototype, MediaFinder is a product of more than a year of development work, including substantial redesign stages. The evolutionary nature of this growth is reflected in the design and the code. Prospects for long-term maintenance and growth of MediaFinder might be improved by a redesign and implementation that accounts for lessons learned to date.

This chapter will provide an overview of MediaFinder implementation along with a description of data flexibility and extensibility of the MediaFinder architecture in terms of Software Engineering.

5.1 MediaFinder Architecture Overview

The MediaFinder acts as a centralized intermediary between user-defined mental models (**Figure 5.1.1**). It also mediates between the database (or file system) and the user-defined mental models. The separation of data, user-defined models and the rest of the system enables the MedaiFinder architecture to be combined with any type of data sets. Theoretically, MediaFinder has been designed to handle both databases and file systems (i.e. each personal media item can be a record in a database table or a file item in a file system) so that a variety of the user-defined mental models can be applied to many types of data sets. This design is originally affected by the paradigm of Model/View/Controller [34]. In the MediaFinder architecture, the personal media data bound to MediaFinder represent a Model, a variety of user-defined models that are used for spatially organizing the data represent a View, and various control panels that are used for specifying the semantics for a *Semantic Region* or formulating a query represent a Controller. MediaFinder's object-oriented architecture can easily be extended to support variants of the *Semantic Regions* Controllers that provide additional expressive power.

In **Figure 5.1.1**, all the bound data sets from either databases or file systems are handled by MediaFinder Data Manager. The Data Manager converts the bound data sets into appropriate data structures to be used for MediaFinder and also update a database or a file system with the personal media items that are newly indexed by MediaFinder. On the other hand, all the user-defined models are handled by MediaFinder Model Manager. The Model Manager controls every data flow between a model and MediaFinder. In addition, the Model Manager also manages all the events triggered by a *Semantic Region* within a model. Furthermore, it handles the events concerning the interrelation between the models if multiple models are combined together on MediaFinder.

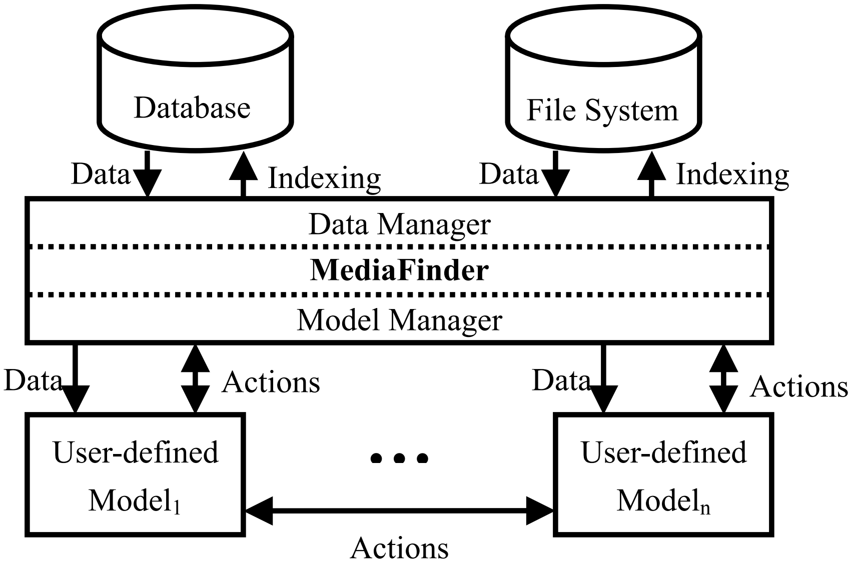


Figure 5.1.1 MediaFinder software architecture

5.2 MediaFinder Code Overview

As an object-oriented programming language application, MediaFinder is composed of several classes. The main project contains a few window form classes needed for the basic operation of MediaFinder:

- frmMainWindow class: The main window class of MediaFinder. It is composed of two panel classes: a Semantics Preview Control Panel class and a Main Panel class. The Semantics Preview Control Panel class includes four different tab windows (folder, date, people, and category) to support flexible hierarchies and to preview the distribution of personal media data based on the collected metadata. The Main Panel class includes two panel windows: a

Semantic Regions panel and a File View panel. The Semantic Regions panel is used as a working space for constructing and using *Semantic Regions* and the File View panel is used for displaying the list of personal media items dynamically based on the metadata that is selected in the Semantics Preview Control Panel.

- frmSemanticOfRegions class: This is a dialog box class used for specifying the semantics for one or more selected regions. This dialog box class is composed of five different tab windows: region, file, date, people, and category. Users can specify the semantics by selecting the metadata collected from personal media items or defining a new one. The specified semantics are converted to a DataContent object and returned from this class when it is closing. The returned DataContent object is referenced by the selected regions to form their semantics. Multiple DataContent objects can be produced by this window class to specify the semantics of multiple regions (bulk semantic specification).
- frmAnnotateItems class: This dialog box class is used for annotating the personal media items contained in a Semantic Region. All or some of the items in the region can be selected and annotated simultaneously with the same attributes specified in the dialog box.
- frmSortRegions class: This dialog box class is used for allowing users to decide the order of bulk semantic specification of the regions located on the MediaFinder main panel. Multiple DataContent objects produced from the

frmSemanticsOfRegion class can have continuously increasing or decreasing attribute values based on the x and y locations on MediaFinder main panel.

- frmProperties class: This dialog box class is used for setting the properties of MediaFinder interaction such as the level of child regions to update, color and size encoding methodology, zooming speed, and so on.

A variety of sub-classes provide the bulk of MediaFinder's data structure and functionality:

- SemanticRegion class: The main semantic region class that inherits the System.Windows.Forms.UserControl class contains the following data members (Appendix A.1).
    - ChildSemanticRegionList: An ArrayList class to store the child *Semantic Regions* objects contained in the current region.
    - PersonalMediaItemList: An ArrayList class to store the references to the Personal Media Items that satisfy the semantics of the current region. Personal media items stored in this ArrayList are searched and collected from the PersonalMediaItemList of the parent region.
    - SelectedSemanticRegionList: An ArrayList to store the references of the selected child regions. This ArrayList is a subset of the ChildSemanticRegionList.
    - IntersectionArray: A two dimensional Array class to keep the information about the containment relationships between the personal media items in the PersonalMediaItemList and the child semantic regions in the ChildSemanticRegionList. This table is

updated whenever either the PersonalMediaItemList or the ChildSemanticRegionList triggers the contents_changed event.

    - IntersectionCountArray: This upper triangular two dimensional Array table is used for counting the shared personal media items among the child regions. Contents of the table are calculated based on the values in the IntersectionArray class and updated every time the IntersectionArray triggers the contents_changed event.
    - BooleanOperandList: An ArrayList class to store the references of sibling regions. The regions in the list are used as operands for the Boolean operation if the current region is created by applying a Boolean operation to the sibling regions. This class checks a region_removed event triggered by the ChildSemanticRegionList of the parent region to make sure whether its operand regions are removed from the ChildSemanticRegionList. Regions are also eliminated from the BooleanOperandList if they are removed from the ChildSemanticRegionList of the parent region.
    - LineList: An ArrayList class to store line objects that represent the line connection between two child regions. This class is monitoring a region_removed event triggered from the ChildSemanticRegionList to remove the broken line caused by the elimination of child regions.
- SemanticContent class: A wrapper class to represent the semantics of a region. Every region is designed to have one SemanticContent class object.

The object contains information about the type of semantics as well as a reference to the corresponding DataContent object.

- FileAttributeInfo class: This class is used for representing the semantics of the regions that deal with files as personal media items. It contains a variety of data members to represent the file properties such as Name, Path, Extension, etc., and three ArrayList objects (Custom, People, and Category) to represent the user-defined semantics (Appendix A.2).
- PersonalMediaItem class: The class to represent the information of each personal media item bound to MediaFinder. It contains five data members that represent the user-defined annotations: people, category, title, subject, and comment. It also contains three data members that stand for the attributes of a file: path, fileinfo (a class object used for storing the file information), and an image object that stores a thumbnail of an image file or an icon of a non-image file and a Boolean tag to represent whether the annotations of the personal media item have been changed (Appendix A.3). This personal media item class is specially designed for a file object, but other types of personal media items (e.g. email item or, a record from database table) can also be used by simply replacing the data members.
- PersonInfo class: This class is used for annotating a person for personal media items. It consists of two character string data members (firstname and lastname) and a few simple member functions for input and output.
- CategoryInfo class: This class is used for annotating a user-defined category for personal media items. It consists of two data members of character

string type (categoryname and categoryvalue) and a few simple member functions for input and output.

- PersonalMediaItemList class: This class is used for storing the list of PersonalMediaItem class objects that are created when the personal media items are bound to MediaFinder either by a static (mapping folder) or by a dynamic (*fling-and-flock*) data binding mechanism.
- Line class: This class is used for representing a line object connecting two regions. This class contains the data members X1, Y1, X2, Y2 to specify the end points of the line as well as data members SR1, SR2 to represent two connecting regions. The color, width, and visibility of the line can also be specified in the class.

5.3 Data Management

5.3.1 Input File Format

MediaFinder uses an xml file format for input files (Appendix A.4). The input file (Semantic Regions Template file) is composed to have a hierarchy. The <Main> tag represents the Main Panel of MediaFinder and it is composed of a <BackgroundImage> and <SRList> tags. The <BackgroundImage> tag is used for a background image of the MediaFinder main panel. The <SRList> tag stands for the list of *Semantic Regions* located on the MediaFinder main panel. The <SRList> tag contains a <Number> tag to represent the number of *Semantic Regions* in the list and the corresponding number of <SemanticRegion> </SemanticRegion> tag pairs. The <SemanticRegion> tag consists of four tags: <SRInfo>, <SRData>,

<ChildSemanticRegions>, and <Lines>. The <SRInfo> tag contains all the physical information about a *Semantic Region* such as the name, x-y location, width, height, background image, shape, type of view, color, text align, font, level, Boolean operation, and the operand regions for the Boolean operation. On the other hand, the <SRData> tag contains all the information about the semantics of the region: file attributes and user-defined metadata such as people, category, title, comments, and subject. Each <People> and <Category> tag can have multiple values. The <ChildSemanticRegions> tag represents the list of child *Semantic Regions* defined inside the region. Just like the <SemanticRegionList> tag in the <Main> tag, <ChildSemanticRegions> contains the <Number> tag to represent the number of child regions and the <SemanticRegion> tag is used recursively to represent the child *Semantic Regions*. With this structure, a *Semantic Region* can have any number of child regions and grandchild regions without depth limitation. The <Lines> tag is placed after the <SRList></SRList> pair or <ChildSemanticRegions></ChildSemanticRegions> pair to represent the list of lines connecting the regions in the <SRList> or in the <ChildSemanticRegions>.

5.3.2 Loading a data file

MediaFinder needs two types of data to be loaded for proper operation: *Semantic Regions* model data and personal media item data. While *Semantic Regions* can be constructed on the fly by users, a previously built model can be saved as a Semantic Regions Template file (*.srt file). A saved Semantic Regions Template file can be retrieved by MediaFinder to reduce users' efforts to construct the similar models over and over again. On the other hand, personal media items should also be loaded and bound to MediaFinder in order to be explored by users.

The process of loading a Semantic Regions Template file begins when users select “Open File…” from the file menu. A System.Windows.Forms.OpenFileDialog class is created and a file name is retrieved through the dialog box. The file name is used to create a SemanticRegionsList object. The member function OpenSRList reads through the metadata at the start of a file and initializes the data structure appropriately. The individual *Semantic Regions* in the data file are retrieved by an OpenSR member function, which is called by the OpenSRList function. OpenSRList creates a new thread which iterates through the file, reading each of the data lines and storing them into a newly created *Semantic Region* and updating the Semantics Control Panel with the semantics specified for each region. When the OpenSRList function finishes reading the data from the file, the RefreshSemanticRegionsWithPMI procedure is automatically called to redistribute the bound personal media items based on newly loaded *Semantic Regions*.

Personal media items are loaded and bound to MediaFinder through a static (mapping a folder) or a dynamic (*fling-and-flock*) data binding mechanism. Once a collection of files is dropped on the MediaFinder main panel, all of the *Semantic Regions,* the contents of the Semantics Control Panel and other data structures are initialized. The AddAllSelectedFiles member function then generates a list of the files and sends it to the InitPersonalMediaItemList function. The InitPersonalMediaItemList function then extracts all the metadata from the files, creates corresponding personal media item objects, and inserts them to the

PersonalMediaItemList. The current version of MediaFinder makes use of Windows NTFS file system to store and retrieve the metadata of personal media data. The NTFS is a file system designed for use specifically with the Windows operating system. It supports long file names, full security access control, file system recovery, extremely large storage media, and various features for the Windows POSIX subsystem. It also supports object-oriented applications by treating all files as objects with user-defined and system-defined attributes. The Windows operating systems such as Windows 2000 and Windows XP can use NTFS and they are able to make use of user-defined attributes called "file summary info" because of an NTFS feature called multiple data streams. Everything that defines an object is passed to the GUI in a data stream. The FAT, Fat32, and VFAT file systems only provide for a single data stream, so the Summary info has to be embedded in the file, as it is with MS Office documents, in order to be passed on. With NTFS, however, additional data may be attached to an object in another data stream entirely. So long as the file is copied from one NTFS volume to another, the information continues to be passed along.

While updating the Semantic Control Panel with the metadata collected from NTFS file system, the function initializes the IntersectionArray and the IntersectionCountArray, and calls the RefreshSemanticRegionsWithPMI function to regroup the newly bound personal media items over the MediaFinder main panel.

5.4 Graphical User Interface

5.4.1 Overall Structure of MediaFinder GUI components

MediaFinder's GUI is implemented as a series of several System.Windows.Forms.Form classes. A schematic overview of the classes involved in the MediaFinder window is given in **Figure 5.4.1**. The frmSemanticRegion is a class inherited from System.Windows.Forms.Form that acts as the main application window. The frmSemanticRegions contains a menu bar, a toolbar and two panel classes: the SRSemanticControlPanel class and the SRMainPanel class.

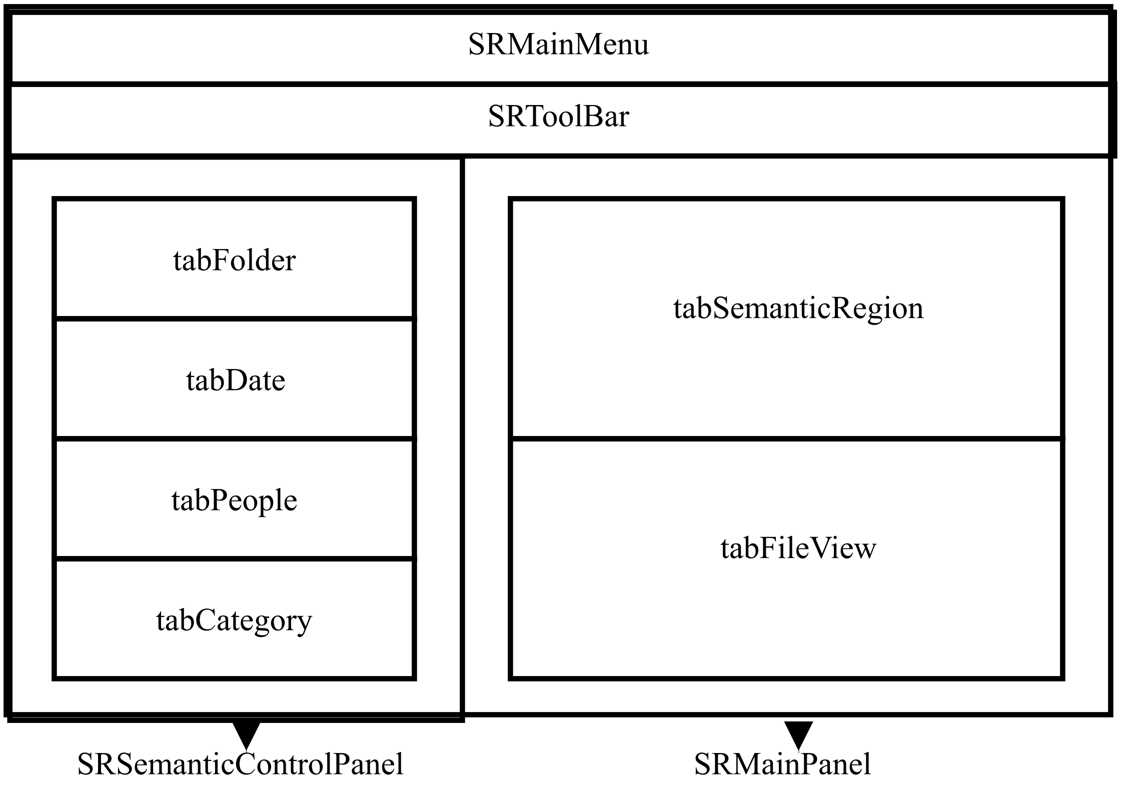


Figure 5.4.1 A schematic overview of the container classes used in the MediaFinder GUI. The entire window is an instance of frmSemanticRegion – an inherited System.Windows.Forms.Form class.

The SRSemanticControlPanel is the panel component located on the left-hand side of the screen. It contains four tab controls (tabFolder, tabDate, tabPeople and tabCategory) and each of them shows four different hierarchies of metadata annotated in the personal media items bound to MediaFinder. Likewise, the SMMainPanel that is located on the right-hand side of application window contains two tab controls: tabSemanticRegion and tabFileView. The tabSemanticRegion panel is a panel component used as a working space for constructing and arranging *Semantic Regions.* The tabFileView panel is a listview GUI component to show the list of personal media items annotated with the selected semantics in the SRSemanticControlPanel.

5.4.2 Interaction Handlers

Semantic Regions and the MediaFinder main panel have their own event handlers to deal with the interactions between users and GUI. These handlers are responsible for every mouse and keyboard event (e.g. MouseClick, MouseDown, MouseUp, MouseMove, MouseEnter, MouseLeave, MouseHover, MouseWheel, KeyDown, KeyPress, KeyUp) produced by users on MediaFinder. There are two different event handlers implemented in the MediaFinder main panel. When users use the toolbar to switch between the *Semantic Regions* creation mode and the *Semantic Regions* modification mode, the SRCreationHandler is activated and the SRModificationHandler is deactivated, and vice versa.

When the SRCreationHandler is active, a mouse press on the main panel leads to the creation of a new *Semantic Region* at the location of the mouse press. On the other hand, when the SRmodificationHandler is active, a click-and-drag mouse action leads to the selection of *Semantic Regions* located on the main panel.

SRmodificationHandler also deals with MouseWheel events to scale the selected regions. Independent to which handler is activated, keyboard events are used to navigate the regions: arrow key events are used to change selection of a region, and enter key and backspace key events are used to zoom in and out of a selected *Semantic Region* respectively.

Drag-and-drop events are processed differently depending on the dragged data. When a DragDrop event is triggered, both handlers check the types of the dragged data. If the dragged data has the same type of personal media items bound to MediaFinder, the event handlers initialize the PersonalMediaItemList with the dragged data and call the RefreshSemanticRegionsWithData function to initiate the *fling-and-flock* interaction. On the other hand, if the dragged data is a user-defined metadata from the Semantics Preview Control Panel, the event handlers construct a new region at the dropped location and specify the dragged metadata for the newly created region. After the construction, the event handlers call the RefreshPartialSemanticRegionsWithData function to gather the personal media items that satisfy the semantics of a newly created region.

Each view of *Semantic Regions* has its own event handler that implements the behavior necessary for that view of *Semantic Regions* (item view, icon view, list view details view, etc.) while modification of all *Semantic Regions* such as moving and resizing is handled by a common handler – the SemanticRegionsEventHandler. SRListViewHandler, SRItemViewHandler, SRDetailsViewHandler, SRAggregationViewHandler, SRIntersectionViewHanlder, and SRBackgroundHandler are the handlers for each view of *Semantic Regions*.

The SemanticRegionsEventHandler supports five main functions:

- Selection of *Semantic Regions*, either via clicking or lasso for group selection
- Moving and resizing of *Semantic Regions*
- Mouse-over highlighting of *Semantic Regions* that share the personal media items (Region Brushing)
- Right mouse click for Semantic Regions popup menu
- Drag-and-drop annotation for Semantic Regions as well as personal media items

When users drag a selected *Semantic Region*, the SemanticRegionsEventHandler calculates its new x-y position and places it to the corresponding location. The resizing handles that are located on the perimeter of *Semantic Regions* can be dragged to scale and this event is also managed by the SemanticRegionsEventHandler. When the mouse is over on the title of a region, the SemanticRegionsEventHandler calls the HighlighSharedRegion method to highlight all the *Semantic Regions* that contain any of the personal media items included in the region. Users can drag a collection of personal media items onto a region to annotate the items with the semantics defined in the region. In addition, metadata can also be dragged from the Semantics Preview Control Panel to a region to specify the semantics. Therefore, when a DragDrop event is triggered, the SemanticRegionsEventHandler first checks the type of dragged data and calls the appropriate methods, either the AnnotatePersonalMediaItems or the AddSemanticsToRegion.

The handlers of the views that show individual items deal with a MouseDown event to initiate the StartDrag event since users can drag the personal media items from a region to other applications for distribution.

5.5 Query Processing

MediaFinder provides dynamic query updates by recomputing query results with every modification to the semantics of any *Semantic Region* involved in the current query. There are several situations where the query results (personal media items contained in *Semantic Regions*) of *Semantic Regions* should be recomputed and thus the personal media items bound to MediaFinder need to be redistributed. These situations can be classified into two categories based on the fact whether the changes of query results of one region affect the query results of other *Semantic Regions*.

The first category, which does not affect the query results of other regions, includes cases such as constructing new regions, importing regions from a *.srt file, copying, and pasting the regions, etc. The process of dynamic query for this category is straightforward because only the newly added regions need to be recalculated and refreshed. However, cases for the second category need to be carefully designed and implemented. This category includes the following two situations:

- The change of query results of a region (either by changing the semantics of a region or by removing the region) that is used as an operand of Boolean operation.
- Metadata of the personal media items are changed through the annotation mechanism implemented in MediaFinder.

If a region is created by applying a Boolean operation to selected regions on MediaFinder, any changes that have occurred in the operand regions should be reflected in the region. However, the region should be updated only after all the operand regions have been updated. The Boolean region (a *Semantic Region* that is created by applying a Boolean operation to *Semantic Regions*) can also be recursively used as an operand region of other Boolean regions. Considering the above two facts, there should exist an order for updating the Boolean regions. Therefore, MediaFinder was designed to have a Boolean operation priority hierarchy to solve this problem as discussed in Section 4.4.5. If all of the *Semantic Regions* have their own priority, the Boolean regions can be easily updated based on the following algorithm (**Table 5.5.1**).

```
Public Sub UpdateBooleanRegions(ByVal SR As SemanticRegion, ByRef List As ArrayList)
    UpdateList.Add(SR)
    SortListByPriority(List)
    Index = List.IndexOf(SR)
    For I = Index To List.Count-1
        SR = List(I)
        numUpdate = UpdatedList.Count
        For J = 0 To numUpdate - 1
            tempSR = UpdatedList(J)
            If SR.BooleanOperands.Contains(tempSR) Then
                SR.ClearData()
                tempSR.PerformBooleanOperation(SR)
                UpdatedList.Add(SR)
            End If
        Next
    Next
End Sub
```

Table 5.5.1 The UpdateBooleanRegion function that updates all of the related Boolean regions starting from the argument region SR based on the priority hierarchy mechanism.

In **Table 5.5.1**, the UpdateBooleanRegion function searches all the sibling regions of the starting region SR in increasing priority order from the priority of SR. If the Boolean operands list of the searched region contains any of the regions in the UpdatedList list, then the Boolean operation of the searched region is forcibly performed immediately and is added to the UpdatedList to show that this region has been updated.

Unlike the first case, there is no good way to partially update the *Semantic Regions* if the annotations of the personal media items have been changed in MediaFinder. Therefore, MediaFinder checks all the first priority *Semantic Regions* (non-Boolean region) whether or not the newly annotated personal media data should be removed from or added to each region. Finally, MediaFinder collects all the regions whose contents have been changed and applies to them an overall updating algorithm similar to **Table 5.5.1**.

5.6 Performance

Information visualization tools strive to provide highly interactive performance for increasingly larger data sets. Although 100ms response time is the goal, this is not always possible depending on the types of data and the size of data. In this section, approximate quantification of the performance of MediaFinder is represented only to provide a rough understanding of its limits. Various sizes of personal media items were tested to evaluate MediaFinder's performance. For each data set, several operations were conducted:

- Constructing a few regions by dragging a metadata tag from the Semantics Preview control Panel

- Dragging the personal media items onto a built-in model to measure the performance of the *fling-and-flock* metaphor
- Applying Boolean operations to the *Semantic Regions*
- Highlighting the shared items among the regions through Region Brushing

For each query, the total processing time including identification of matching items and all screen updates was measured. The values were averaged after a few repeated experiments for an average query processing time for each task and for each data set.

Data sets with 100, 500, 1000, and 5000 personal media items (all of them are image files) were tested. MediaFinder was unable to handle a data set with 10000 items since this data set exhausted available main memory on the test computer. However, MediaFinder was able to handle a test data set with more than 20000 non-image items without any problem because it did not need to use that much main memory in loading the file icons as opposed to loading the thumbnails of image files.

All tests were run on a 1.6GHz Pentium-M with 640MB main memory and 32MB video memory running Windows XP. Average response times for the above queries are given in **Figure 5.6.1**. The results of the second test (*fling-and flock* the items over the built-in model that is composed of five *Semantic Regions*) show that MediaFinder's performance of *fling-and-flock* interaction scales linearly with the number of personal media items. These results can be used to generate a regression that would predict the performance of *fling-and-flock* on data sets of various sizes.

It is interesting to note that the other three tests (Drag-and-drop region creation, Boolean operation, and Region Brushing) do not scale linearly with the number of personal media items. For the first test (Constructing a *Semantic Region* by dragging a metadata tag from the Semantics Preview Control Panel), it is presumed that the majority of time was spent in creating a new Semantic Region while searching is a relatively small part of the overall cost for this type of query processing. On the other hand, the results of the third test (Constructing a Boolean region by applying a UNION operation to two regions) shows that the query performance depends more on the total number of items contained in the regions to which a Boolean operation has been applied than the total number personal media items bound to MediaFinder. Like the results of the third test, the fourth test (Highlighting the shared items among regions by placing the mouse over the title of a region) is more likely dependent on the number of items in the query region (the region that the mouse is placed), the number of sibling *Semantic Regions* and the average number of items contained in the sibling regions, rather than the total number of items bound to MediaFinder. However, because those factors are somewhat correlated to the total number of personal media items with real data sets, the overall query performance for each test seems to have a positive correlation with the total number of personal media items bound to MediaFinder.

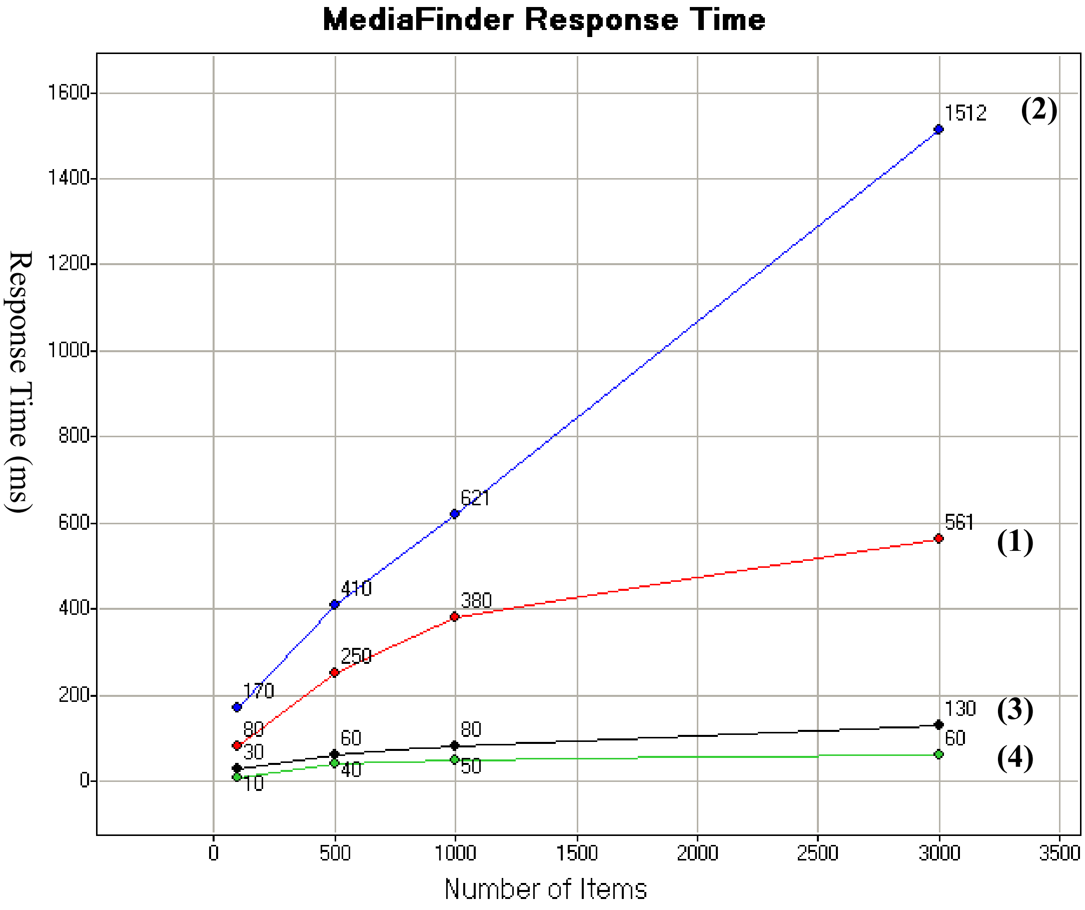


Figure 5.6.1 Average times for MediaFinder to completely process queries – including search and display update – on several query types. Results are shown for data sets of 100, 500, 1000 and 3000 items with 4 different query tests. (1) Constructing a *Semantic Region* by dragging a metadata tag from the Semantics Preview Control Panel (2) Dragging the personal media items onto a built-in model to measure the performance of the *fling-and-flock* metaphor (3) Constructing a Boolean region by applying a UNION operation to two regions (4) Highlighting the shared items among regions by placing the mouse over the title of a region

Despite the high correlations, these results are fairly limited in their applicability and generality. The tasks used to generate these results were not rigorously controlled, and the specific timing values are not generalizable beyond the computer system that was used to run the test. However, the results provide a rough measure of the scalability of MediaFinder. With the computer used in this test, 100ms performance is only likely to be possible for the Region Brushing query on data set less than 10,000 items. However, considering general search tasks such as searching a file from hard disk or searching a webpage using web search engines, performance of MediaFinder is not unreasonably slow. As performance continues to increase, performance for MediaFinder queries with data sets containing 10,000 personal media items within a second may soon be possible.

5.7 Extension

While the MediaFinder software architecture is currently implemented as a standalone application, it could be integrated into database systems or operating systems. For example, MediaFinder could be integrated into the file system architecture in the Windows operating system. The MediaFinder could substitute for the Windows file explorer with additional features such as spatial organizations of file items based on users' mental models, dynamically generated file hierarchies based on user-defined categorizations, file search by combining user-defined semantics, file indexing with user-defined semantics, etc., while still providing the feature of browsing the file-folder hierarchy, which the Windows explorer mainly does.

On the other hand, MediaFinder could be integrated into database systems as a front end GUI for data queries. While MediaFinder is unable to provide the full power of SQL, it could provide users a quick overview of data in terms of their preferred mental models and help users perform quick and simple queries so that the users can understand data more easily and intuitively.

Figure 5.7.1 shows an example of MediaFinder extension to a database table. Since MediaFinder has been designed based on the Model/View/Controller [34] paradigm, which was first introduced in Samlltalk 80, MediaFinder can handle various types of personal media data in practice. In this example, MediaFinder was bound to a database table that represents a Natality data of DC in 2000. The table contains 11639 records and each record represents various information about the mother (24 attributes) of a new born baby such as place or facility of birth, age of mother, race of mother, education of mother, marital status of mother, adequacy of care, and so on.

Unlike other types of personal media items, the records in a database table were bound to MediaFinder through the Windows clipboard. In other words, the selected records in the table were copied and pasted to the MediaFinder main panel to be bound to MediaFinder. During the data binding process, MediaFinder collected all the attributes and their values from the clipboard, and showed them in the Semantics Preview Control Panel. In **Figure 5.7.1**, five attributes were chosen (adequacy of care, education of mother, race of mother, race of father, and marital status of mother attributes) from the Semantics Preview Control panel and the regions corresponding to each attribute value are constructed on the main panel. The constructed regions were resized proportional to the number of items they contained. In **Figure 5.7.1**, a tiny square that is contained in each region represents a record satisfying the semantics of the region. By combining the five different attributes at once, users can figure out some interesting facts concerning the interrelationships between the attributes through Region Brushing. For example, just by placing a mouse over the title of the unmarried mother region, users can easily extract meanings from the model such as, “more than a half of the intermediate or inadequate cares came from unmarried mothers”, “more than half of the lowly educated mothers were unmarried”, “about 3 quarters of black mothers were unmarried", "almost all of fathers whose race was not stated or unknown were unmarried", and so on.

MediaFinder can be compared to TableLens [79] in terms of visualizing the interrelationships between the different attributes of tabular data, but their approaches are fairly different. In TableLens, each attribute value of a record is converted to a histogram bar and all the records in a data table are sorted by the value of the selected attribute. Users can recognize the correlations between the selected attribute and other attributes by observing the changing patterns of histogram bars with respect to one selected attribute. On the contrary, MediaFinder handles a record as an object that can be contained in multiple regions whose semantic represents an attribute value of data table. Users can recognize the interrelationship between the attributes by observing the number of or the percentage of the shared records between the regions. In other words, TableLens focuses on visualizing the correlations between the attributes of the records through the sorted histogram and is suitable for answering the questions such as, "Is there any correlation between the average number of cigarettes per day and average number of drinks per week in DC natality data?", "Is there any correlation between the birth weight detail in grams and age of mother?", "Is there any correlation between age of father and age of mother?", and so on. On the other hand, MedaiFinder focuses more on visualizing the distribution of data records based on each attribute value in the data table and is appropriate for answering the queries like, "What percentage of black mother are unmarried?", "What percentage of white mothers have white fathers?", What percentage of highly educated mothers are married?", and so on. MediaFinder is also good for dynamic and hierarchical regrouping of the records based on the various attributes. For example, the tasks such as "Show the age distribution of black mothers" or "Show the distributions of race of mothers with respect to the adequacy of care" require regrouping of the records containing a certain attribute value by another attribute, which are not easy to accomplish with TableLens.

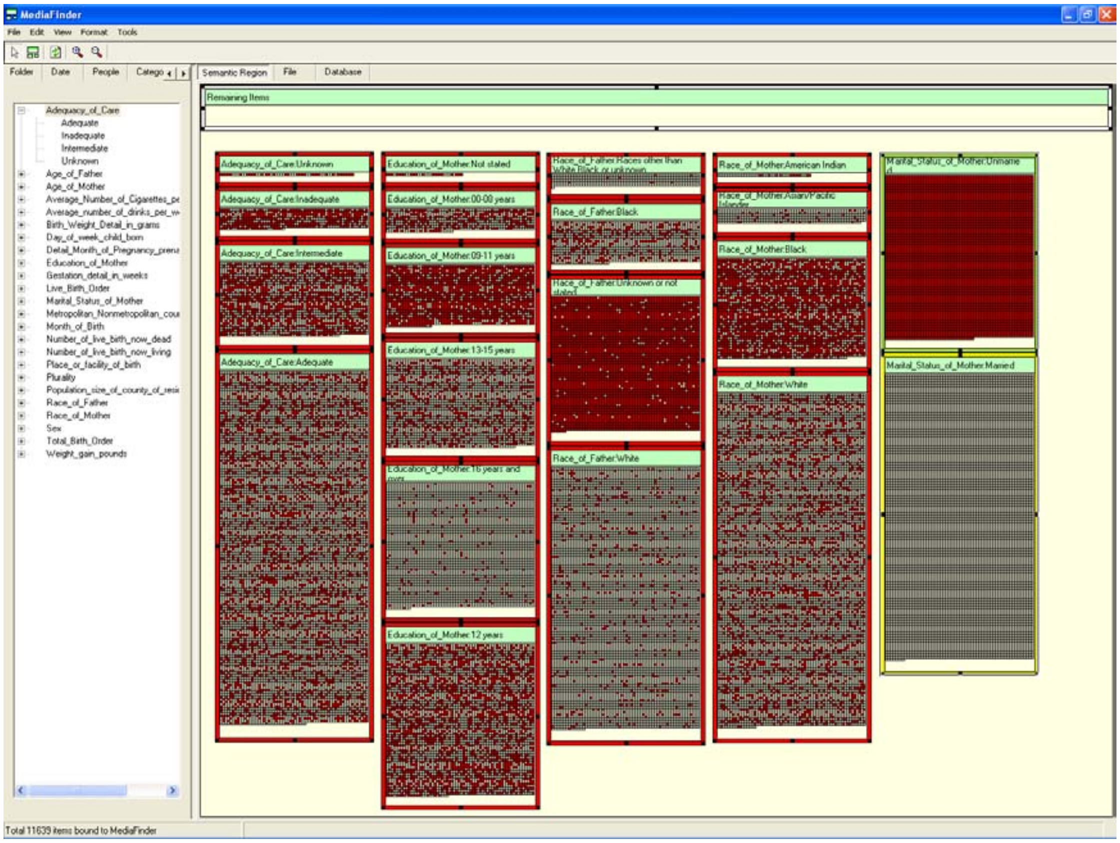

MediaFinder
File Edit View Format Tools
Folder Date People Catego
Semantic Region File Database
Adequacy_of_Care
Adequate
Inadequate
Intermediate
Unknown
Age_of_Father
Age_of_Mother
Average_Number_of_Cigarettes_pe
Average_number_of_drinks_per_w
Birth_Weight_Detail_in_grams
Day_of_week_child_born
Detail_Month_of_Pregnancy_prena
Education_of_Mother
Gestation_detail_in_weeks
Live_Birth_Order
Marital_Status_of_Mother
Metropolitan_Nonmetropolitan_cou
Month_of_Birth
Number_of_live_birth_now_dead
Number_of_live_birth_now_living
Place_or_facility_of_birth
Plurality
Population_size_of_county_of_resi
Race_of_Father
Race_of_Mother
Sex
Total_Birth_Order
Weight_gain_pounds
Remaining Items
Adequacy_of_Care:Unknown
Adequacy_of_Care:Inadequate
Adequacy_of_Care:Intermediate
Adequacy_of_Care:Adequate
Education_of_Mother:Not stated
Education_of_Mother:00-08 years
Education_of_Mother:09-11 years
Education_of_Mother:13-15 years
Education_of_Mother:16 years and more
Education_of_Mother:12 years
Race_of_Father:Races other than White Black or unknown
Race_of_Father:Black
Race_of_Father:Unknown or not stated
Race_of_Father:White
Race_of_Mother:American Indian
Race_of_Mother:Asian/Pacific Islander
Race_of_Mother:Black
Race_of_Mother:White
Marital_Status_of_Mother:Unmarried
Marital_Status_of_Mother:Married
Total 11639 items bound to MediaFinder

Figure 5.7.1 The Extension of MediaFinder to database table. In this example, five different attributes (adequacy of care, education of mother, race of mother, race of father, and marital status of mother attributes) were joined together to visualize the interelationships between the attributes. All the records representing unmarried mother were highlighted through the Region Brushing feature to visualize the distributions of unmarried mother with respect to four other attributes.

The extension of MediaFinder has a few major benefits. Both operating systems and database systems benefit by adding this powerful new feature. MediaFinder benefits by potential improved performance due to integration. This approach might also lead to a new design of file systems so that they can handle the high-level user-defined attributes other than the existing low-level file attributes.

In summary, MediaFinder software architecture enables flexibility in data and visualizations. Its separation of data and visualizations enables users to easily manage various types of data based on their preferred mental models. The data structure and algorithms are based on the sound *Semantic Regions* model. MediaFinder provides a solid foundation for a potent new future direction.

Chapter 6:

Evaluating the Use and Construction of *Semantic Regions*

Studying the use of Semantic Regions is important for two reasons:

- To evaluate the usability and benefits of MediaFinder based on Semantic Regions and discover potential user interface improvements.
- To gain a deeper level of understanding about users' ability to understand, construct, and operate Semantic Regions strategies in general.

Two separate studies were undertaken to evaluate two distinct aspects of Semantic Regions.

1. **Use:** First, can users understand the concept of *Semantic Regions* and mental models to manage and explore their personal media data?
2. **Construction:** Can users successfully construct *Semantic Regions* based on their own mental models toward their personal media data?

6.1 Evaluation of Semantic Regions Use

The goal of this study is to determine if users can learn to use the MediaFinder interface as well as the predefined *Semantic Regions* templates and how difficult it is for users to use them, in terms of success rate and time to completion and to identify cognitive trouble spots in the process. Hence, this study examines the flexibility that MediaFinder provides. Can users grasp the concept of organizing personal media items according to the given templates? What cognitive issues are involved, how much training is required, how do users' backgrounds affect performance and can relatively novice users use MediaFinder in managing their personal media data in a short time? This study also reveals potential MediaFinder user interface improvements.

Therefore, the following variables were measured during the test:

- Subjects' background information
- Learning time
- Success (y/n how close to success)
- Time to completion

This study also observed:

- Cognitive trouble spots (in training and test trials).
- MediaFinder interface problems.

6.1.1 Procedure

A total of five subjects participated, one at a time. Two of them were native English speakers and three of them were not. They were each paid 10 dollars for participating in the user study. Their respective majors are Nutrition & Food Science, Hearing & Speech Science, Civil Engineering, Criminology and Art studio. They were not highly experienced personal computer users in terms of computer usage experience, programming experience, and the number of applications they use on their machines. They all use Windows operating system both in the office and at home (none of them are Macintosh users).

First, background information was obtained from each subject concerning their experience with: computers, Windows file system, object-oriented drawing tools, visualization tools, programming, personal media, digital photos, internet access, and so on (Appendix B.1). **Table 6.1.1** shows more details about the computer experience of the participants. They have been using computers for 6~16 years since they took computer 101 courses during either high school or college and they use computers mainly for word processing, internet (email and web browsing) access as well as using a few applications related to their respective fields. Since all the subjects were MS-Windows users, they were familiar with the Windows file-folder hierarchy concept and knew how to create a folder, how to organize a folder hierarchy, how to browse folders, how to delete, copy, and move files from disk, and so on. All the subjects have experienced at least one object-oriented drawing tool such as MS PowerPoint, Adobe Illustrator, Snagit Studio (www.techsmith.com), Paint Shop Pro (www.jasc.com), etc., even though they didn't know what the term "object-oriented" means. They all had experience with only basic visualization tools such as Excel charts, statistical package SPSS graphs, and some chart tools in their commercial software. For programming experience, two subjects have not used any programming language at all, and the others said that they took one or two programming language introduction courses (FORTRAN, BASIC and C-language) during their undergraduate years but they have not had a chance to program any code after that.

Category	**Subject 1**	**Subject 2**	**Subject 3**
Gender	Female	Female	Female
Age	31	26	25
Major	Nutrition and Food Science	Hearing and Speech Science	Civil Engineering
Computer usage experience	10 years	6 years	6 years
Main Usage	Word processing, Internet	Word processing, Spreadsheet, Internet	Word processing, Scientific Calculation, Internet
Window file system concepts	Familiar	Familiar	Familiar
Object-oriented drawing tool experience	Novice	Novice	Novice
Visualization tool experience	Basic (Excel chart)	SPSS, Excel chart	Sigma Graph, Excel chart
Programming Experience	1 year	None	1 year
Digital photo experience	Novice	2 years	2 years
PhotoFinder Experience	None	None	None
Frequently used Photo Browser	Explorer	ACDSee	Explorer
Types of personal media data stored in personal computer	Office documents	Movies, Photos, Music, Ebooks	Movies, Photos, Music
Home Internet Access	Dialup modem (56 Kbps)	Cable Modem (1.5 Mbps)	Cable Modem (1.5 Mbps)
Internet (hrs/day)	2	3~4	3~4

Table 6.1.1. Background information of 5 subjects participated in the first user study

Category	**Subject 4**	**Subject 5**
Gender	Female	Male
Age	21	29
Major	Criminology	Professional Artist (Art Studio)
Computer usage experience	7 years	16 years
Main Usage	Word processing, Games, Internet	Word processing, Image editing, Internet
Window file system concepts	Familiar	Familiar
Object-oriented drawing tool experience	Novice	Novice
Visualization tool experience	Basic (Excel chart)	Rare
Programming experience	None	One semester (C-language)
Digital photo experience	No	One and a half years
PhotoFinder Experience	None	None
Frequently used Photo Browser	Explorer	Explorer Photosuite
Types of personal media data stored in their personal computer	Office documents	Movies, Photos, Music
Home Internet Access	Dialup modem (56 Kbps)	ADSL (256 Kbps)
Internet (hrs/day)	6~7	1~2

Table 6.1.1 Background information of 5 subjects participated in the first user study (Continued)

Then each subject was trained on MediaFinder. The 10-minute training consisted of:

1. A quick demonstration of MediaFinder by the administrator in order to give the subject an overview and motivation. (5 minutes)
2. Review of various personal media management concepts such as organization, meaning extraction, search, and navigation. (2 minutes)
3. Detailed instruction on the usage of MediaFinder and Semantic Regions. The subjects walked through the usage of Semantic Regions for managing and exploring personal media data. This demonstrated how to group and regroup personal media based on the Semantic Regions templates using the *fling-and-flock* metaphor, how to navigate the hierarchy of regions, how to search for items, and how to extract meanings from the data set. (3 minutes)

Then, when confident to continue, each subject began the testing phase. First, subjects were given a set of practice tasks in order to get accustomed to the MediaFinder interface. During the practice tasks, they were allowed to ask any questions about MediaFinder interface and background knowledge.

- Practice Tasks:
  - ■ Open a Conference Calendar model
  - ■ Can you use a *fling-and-flock* metaphor to a given template?
  - ■ How many pictures were taken in CHI 2000?
  - ■ Can you select multiple regions with mouse?
  - ■ Can you change the view of multiple regions?
  - ■ Maximize a region and browse the photos contained in it. Can you see a

full-size image?

- Minimize the region.

After the completion of the practice tasks, they were given a set of tasks along with a test data set. The test data set consisted of 281 photos that were taken at the CHI conferences between 1991 and 2000. The photos were stored in one directory and their metadata (e.g. location, date, and people) was annotated with PhotoFinder4 (**Figure 6.1.1**). In addition to photos, four Semantic Regions templates were given:

- Location Map (7 States and 8 Cities in US) (**Figure 6.1.2**)
- Conference Calendar (10 Years of Calendar) (**Figure 6.1.3**)
- People (9 Most frequently appearing people) (**Figure 6.1.4**)
- Combined Model (Combination of the above three models) (**Figure 6.1.5**)

To simplify the questions and focus only on the main concept of Semantic Regions, the questions about navigation (more than two levels), distribution, and annotation have been omitted. A total of 7 revised tasks were given to the users after short descriptions on the data set and Semantic Regions templates.

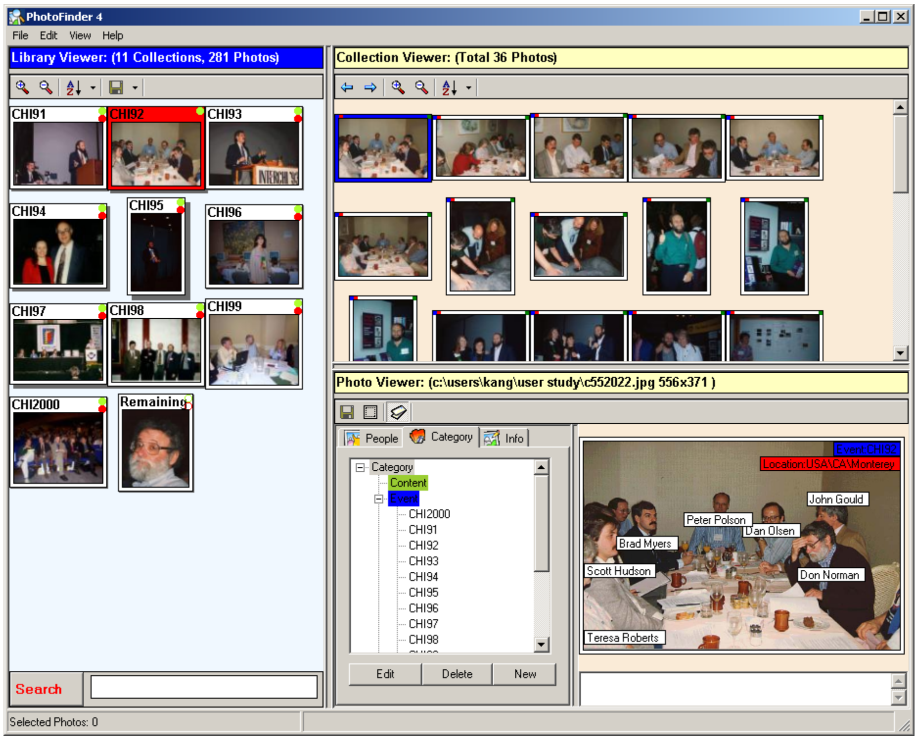


Figure 6.1.1 User study data set: a total of 281 photos chosen from the CHI photo library (from CHI91 to CHI2000). Metadata about location, conference name, date, and people have been combined to photo files using PhotoFinder4

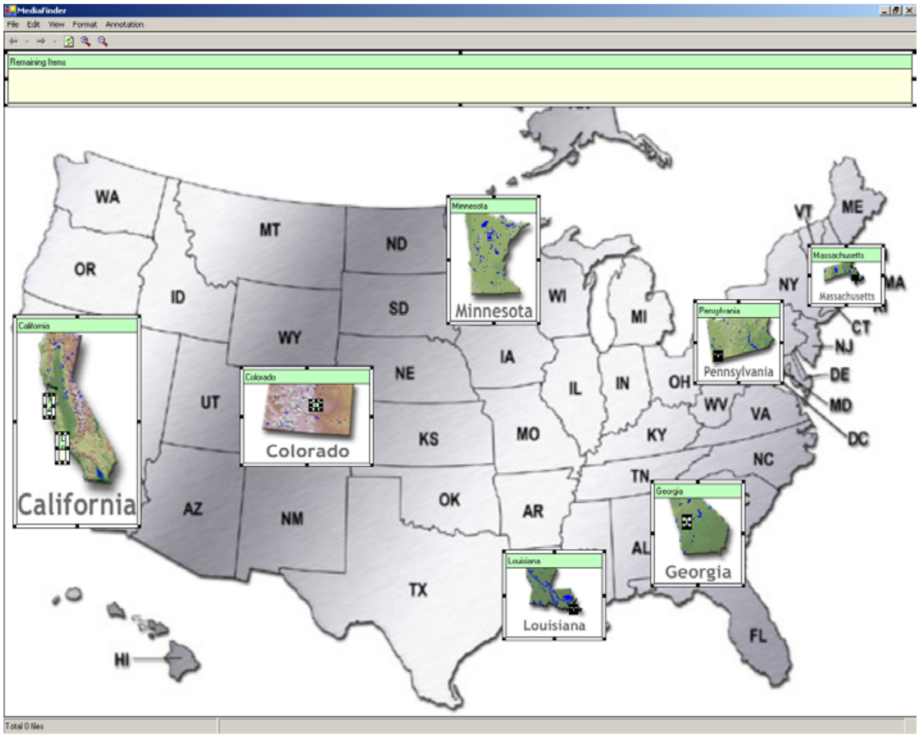


Figure 6.1.2 US map mental model with 7 state Semantic Regions and 8 cities of child regions.

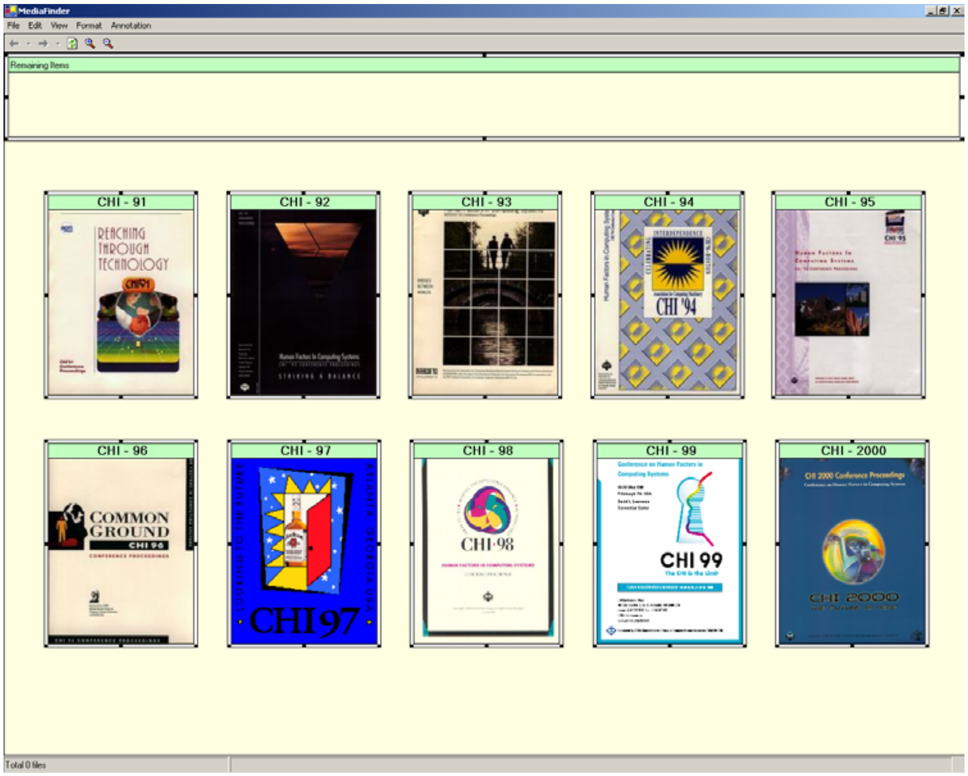


Figure 6.1.3 Conference calendar mental model: each region represents a CHI conference between CHI91 and CHI2000.

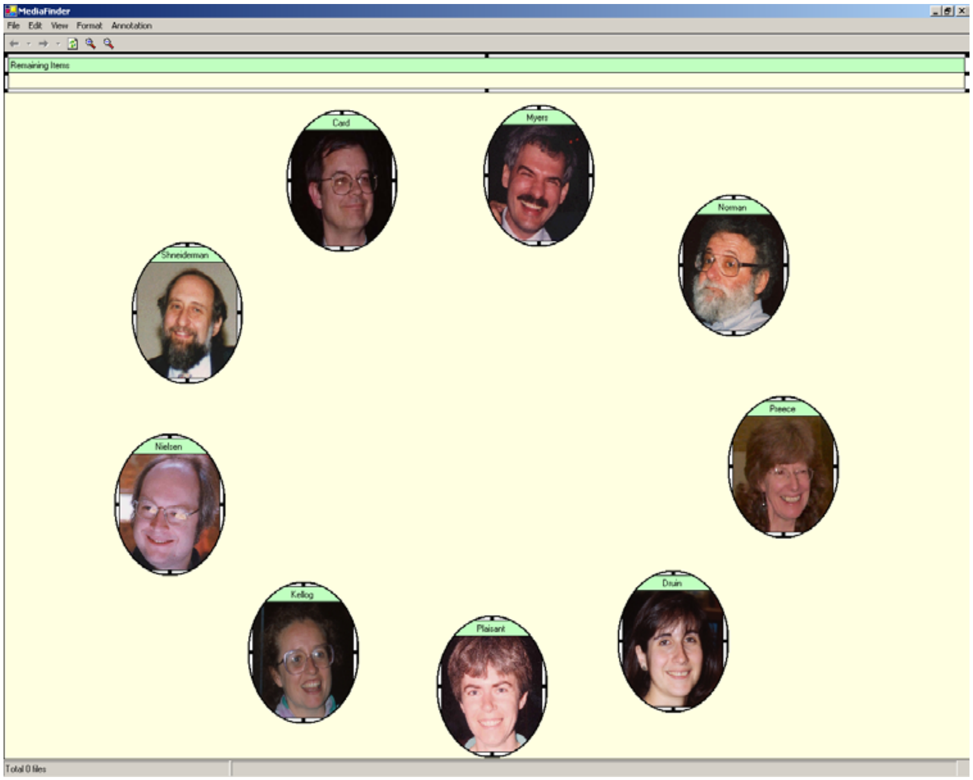


Figure 6.1.4 People mental model: the 9 most frequently appearing people from the test data set were chosen.

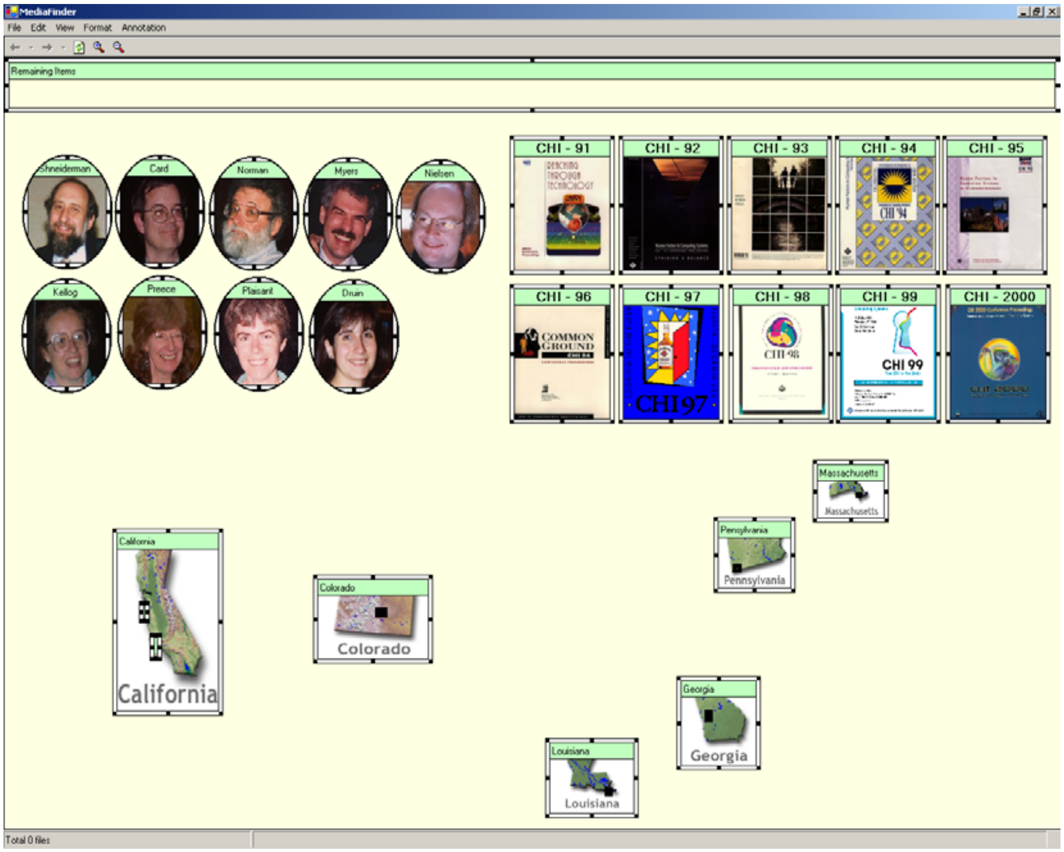


Figure 6.1.5 Combined-model (People, calendar, and location)

7 real tasks

- **Location Model**

1. Group the photos based on the US map. Which state has the most photos and which state doesn't have any photos? How many photos were taken outside the US? (Organizing photos using the *fling-and-flock* metaphor)

- **Calendar Model**

2. Regroup the photos based on the Conference Calendar model. How many photos were taken in each conference?

   (Dynamic regrouping of personal media: Apply a different mental model to the same data set.)

- **People Model**

3. How many people appeared in at least one photo along with Ben Shneiderman?
   (Understand the multiple existence of an item in the regions)
4. Find all the photos where both Shneiderman and Kellogg appeared.
   (Search: apply Boolean operations to the regions)

- **Combined-model**

5. Find the name of the conferences that all nine people participated in.
   Where (what is the name of the state) was the conference held?
   (Find out the relationship among the different mental models)
6. What was the name of the conference held in Atlanta, Georgia?
   Find out who didn't appear in the photos taken at this conference.
   (Change the focus among different models)
7. Find the conferences that were not held in the US.
   (Extra meaning extraction task: Observe the users' various approaches to solve the problem)

Finally, subjects were given the opportunity to freely explore the system, describe the problems with the MediaFinder interface, and offer suggestions for improvement through the following post verbal survey (Appendix B.2).

6.1.2 Background Information

From the background survey, none of the subjects was a highly experienced personal computer user in terms of computer usage experience, knowledge about computer systems, and programming experience. Each had either none or limited experience with basic visualization tools (e.g. charts and graphs). They managed a few types of personal media data such as video clips, audio clips, photos, and books on their machines, even though the amount of personal media data was not so large. **Table 6.1.2** represents the types and the amount of personal media data they had on their machines. Three of the subjects had their own digital cameras and they had 1.5 ~ 2 years of digital photo experiences. The other two subjects managed pictures scanned and collected from internet. Only one subject was using a photo browsing tool (ACDSee) and the others used the default file explorer in Windows to browse photos or other personal media they had. None of the subjects have ever made any kind of annotation for their media data except the name of a folder. They had not used any personal media management tools either. They all depended on their memory to search and browse the personal media data they had in storage.

	Subject 1	**Subject 2**	**Subject 3**	**Subject 4**	**Subject 5**
Video clips	None	About 50	About 50	None	About 30
Music clips	None	About 30	About 200	None	About 30
Photos	None	About 1,000	About 300	None	About 400
Ebooks	None	About 20	None	None	None

Table 6.1.2 The types and the amount of personal media data that pilot study users have.

6.1.3 Result

For the actual tasks, all the subjects were able to complete all 7 questions. During the tests, subjects 3 and 4 asked for help in reminding them how to apply Boolean operations for task 4. On the other hand, subject 5 asked for help in understanding the meaning of the highlighted regions for task 3. All tasks were successfully completed. In terms of the time it took for users to complete the given tasks, it took less than 3 minutes for each subject to complete each task except for those tasks which required the assistance of the administrator (see **Table 6.1.3**).

After a 10 minute introduction of MediaFinder usage and concepts, they understood most of the features and were ready to use MediaFinder. In general, the subjects were quick to learn the concepts and usage and were capable of using the Semantic Regions templates to answer the given tasks. They showed interest in the *fling-and-flock* metaphor and the concept of dynamic grouping of media items. It seemed that they preferred (more easily understood) dragging the file items to dragging a folder containing file items, but as soon as they got accustomed to the *fling-and-flock* metaphor they started dragging a folder in order to eliminate a step for selecting file items.

	Subject 1	**Subject 2**	**Subject 3**	**Subject 4**	**Subject 5**
Task 1	S (2 m)	S (30 s)	S (50 s)	S (1m 40s)	S (2m 50s)
Task 2	S (1 m)	S (10 s)	S (20 s)	S (20 s)	S (30 s)
Task 3	S (1 m)	S (15 s)	S (1m 10s)	S (1m 10s)	SH (3m 30s)
Task 4	S (30 s)	S (20 s)	SH (3m 30s)	SH (1m 30s)	S (15 s)
Task 5	S (2 m)	S (20 s)	S (15 s)	S (20 s)	S (2 m)
Task 6	S (10 s)	S (10 s)	S (10 s)	S (15 s)	S (15 s)
Task 7	S (1 m)	S (10 s)	S (20 s)	S (1 m)	S (2 m)

Table 6.1.3 Task success (S: success, F: fail, SH: success with help) and time to completion (m: minute, s:second)

They did not seem to have any problem in understanding the concept of *Semantic Regions* when given the explanation that "each region is just like a folder and each media item can be considered as a file item." In addition, they easily understood that the concept of child regions inside a region was just like a folder hierarchy structure. All the subjects mentioned that they were able to more easily understand the given model with the background images (e.g. maps of states, pictures of people, or calendar) of MediaFinder and each region.

They were able to regroup the items by applying a different *Semantic Regions* template. However, in the "People" model, three of the subjects were slightly confused about the fact that each photo item could be contained in multiple regions. But with a short explanation about why this happened and how it could be used to extract meanings, they were not confused anymore. With the understanding of the multi-existence of items, the subjects had no difficulty in applying Boolean operations to the selected regions and searched the items as requested. At first (during introduction), none of the subjects understood what the word "Boolean operations" meant, but when they were given explanations about "Venndiagrams" together with the intersection, union, and complement operations, they were able to understand the concept without any problem.

In the combined model (combination of People, Calendar, and Location), it was surprising that it took much less time to complete the task than it took for the single models. They did not seem to have any serious problem in understanding the combined model and they were eager to find out more interesting facts beyond the tasks that they were asked to answer. For the last task ("Find the conferences that were not held in the US."), all the subjects tried to answer the question by positioning the mouse cursor on the title of each conference calendar and observed whether there were any highlighted US state regions in the Location model. When they were asked to find another way to solve this task, three of the subjects figured out that they could select all the US state regions and apply union operations to them, and then observe whether there were any non-highlighted regions in the Conference Calendar model. Although the rest of the subjects did not know how to do that, they did understand the meaning when the solution was shown.

In the verbal post survey, all the subjects showed their interests and they were eager to use the *fling-and-flock* metaphor for other applications. One subject said that she usually had trouble finding figures, tables, references, and related materials for each chapter of a paper from the hard disk and that she would like to use the *fling-and-flock* metaphor to organize her paper materials and to use MediaFinder as a paper material management tool for her paper writing. On the other hand, the professional artist mentioned that he would like to use MediaFinder for organizing his paintings dynamically. He said that he had a lot of paintings and always took pictures of them for future display and that he wanted to group them based on various categories such as color, painting medium, canvas size, timeline, exhibition location, and so on. He said that it was crucial for him to show his buyers sample paintings conveniently based on their tastes and that he was eager to use MediaFinder as his painting organizing tool.

For the questionnaire concerning ease of use (scale 1 to 9) of the MediaFinder interface, they marked an average of 8.2 points (See **Table 6.1.4**) and stated that the

MediaFinder interface was not particularly difficult for them to use compared with other interfaces or the Windows file explorer. They all agreed that the first 10-minute introduction helped them understand the concept and main features, but without it, they would not have been able to answer the questions.

They also commented on the MediaFinder interface features which were confusing during the test:

- Two of the subjects did not realize that the border of a region has three colors; (white: unselected region, yellow: selected region, and red: highlighted region). They said that they recognized the selected regions just by a bounding rectangle drawn over the selected regions.
- One subject mentioned that she thought that the regions were supposed to be highlighted when the mouse was over the body of the region, not on the title of the region.
- Two of the subjects mentioned that they were a little confused as to where the right mouse button should be clicked in order to bring up a popup menu for changing the view of multiple selected regions. (click on the region? or click on the empty background of MediaFinder?)
- One of the subjects was embarrassed when nothing happened after she dragged the items on the MediaFinder. (That was because she dropped the items onto the region and not onto the empty background of MediaFinder) She expected that she could drop the items anywhere on the interface with *fling-and-flock*, but as a matter of fact, dropping items on a region is used for annotating the dropped items with the semantics defined in the region.

- One of the subjects mentioned that when she selected two regions and moved the mouse over their titles, she expected that only the regions that contained the photos existed in both of the selected regions were highlighted, so she did not try to apply an intersection operation to complete task 4.
- Make the highlighted regions more visible e.g. making non-highlighted regions translucent.
- Views of a region do not have consistent reactions for a double-click event (e.g. in details view, the double click brings up the associated program, but in an aggregation view, the double-click event maximizes a region)

	Subject1	**Subject2**	**Subject3**	**Subject4**	**Subject5**
Ease of use	8	9	8	7	9
Willingness to use (with given metadata)	9	8	9	9	9
Willingness to use (with manual annotation)	6	3	3	3	5

Table 6.1.4 User satisfaction survey of MediaFinder (1 – 9 scale)

For the questionnaire about the willingness to use (scale 1 to 9), they gave an average of 8.8 points under the condition that all the metadata are given and not annotated by themselves. All the subjects wondered where the photo metadata came from and they were very concerned about the annotation process. They reported that they were not likely interested in using MediaFinder unless the metadata are automatically extracted or collected from the photos and they marked only an average of 4 points for "willingness to use with manual annotation." After the test, the PhotoFinder4 beta version was shown to everybody especially to highlight its features for creating a user-defined category tag, *bulk annotation*, and *direct annotation*. All of the subjects mentioned that they would like to use MediaFinder if MediaFinder had a similar kind of annotation feature.

Overall, all of the subjects (even though they were inexperienced users) did not seem to have any serious problems grasping the cognitive concept of *Semantic Regions* and using the MediaFinder interface for managing personal media. They liked the concepts of spatial organization and dynamic grouping of the personal media data based on the different models.

6.2 Evaluation of Semantic Regions Construction

The goal of this study is to determine if users can learn to construct *Semantic Regions* and how difficult it is for users to construct them and to identify cognitive trouble spots in the process. As in the first user study, the second study will focus on the following questions; Can users grasp the concept of specifying semantics as well as constructing Semantic Regions? What cognitive issues are involved, how much training is required, how do users' backgrounds affect performance, and can relatively novice users construct Semantic Regions based on their mental models in a short time? This study also reveals potential MediaFinder user interface improvements.

MediaFinder is used to examine these issues. Currently, MediaFinder employs a 2-step approach to construct Semantic Regions. First, users construct 2 dimensional rectangular regions on the MediaFinder interface. Second, users specify the semantics for each region and bind the personal media data with the constructed model using the *fling-and-flock* metaphor.

6.2.1 Procedure

A total of five subjects participated, one at a time. Three of them were computer science graduate students on campus and they were expert users of personal computers as well as MS Windows (none of them were Macintosh users). They were highly experienced users in terms of computer usage experience, programming experience, and the number of applications they use on their machines. On the other hand, two of them were not experienced users that were chosen from the first user study. All the subjects already knew how to use *Semantic Regions* because they participated in the first user study either as pilot subjects or as actual subjects. This was intended to reduce the users' efforts to learn both how to use *Semantic Regions* and how to construct them. Just as in the first user study, all the participants were paid 10 dollars for participation.

Background information was obtained from the three experienced users concerning their experience with: computers, Windows file system, object-oriented drawing tools, visualization tools, programming, personal media, digital photos, internet access, and so on. This step of collecting the users' background information was omitted for two novice users because it was already done in the first user study. **Table 6.2.1** shows the details about the computer experience of the participants (3 experienced users only). They frequently use their personal media data for entertainment such as playing movies, listening to music, and browsing photos. One of the subjects had a large amount of personal media data: about 25,000 comic books (scanned images in 590 CDs), 800 movie CDs, 5000 electronic books (most of them are text format), 1,000 music (mp3) files, 6000 digital photos, and about 300 digital video files including documentary films and TV shows. On the other hand, another subject only had about 30 movies, 400 mp3 files, and 1,000 photos. They all had high speed internet access at home such as a DSL and cable modem and most of their personal media data were collected from the internet (e.g. p-to-p file sharing system, ftp, http, etc.). Only a small amount of the data was created by themselves except for digital photos and home video files.

Each subject was trained on MediaFinder. The training program consisted of:

1. A quick demonstration of constructing Semantic Regions by the administrator to give the subject an overview and motivation. (3 minutes)
2. Detailed instruction on how to construct Semantic Regions and define semantics. The subjects will walk through the usage of MediaFinder for Semantic Regions Construction. This will demonstrate how to create regions, how to resize and reposition them, how to delete them, how to specify semantics for them, how to save and load a model, how to combine the predefined models, and so on. (7 minutes)

Then, when confident to continue, each subject began the testing phase.

First, subjects were given a set of practice tasks in order to make sure that they understood the demonstration correctly. During the practice tasks, they were allowed to ask any questions about the MediaFinder interface as well as background information.

Category	**Subject 1**	**Subject 2**	**Subject 3**
Gender	Male	Male	Male
Age	34	29	38
Major	CS grad student	CS grad student	CS grad student
Computer usage experience	15 years	17 years	20 years
Window file system concepts	Familiar	Familiar	Familiar
Object-oriented drawing tool experience	Expert	Intermediate	Expert
Visualization tool experience	Rare	Rare	Rare
Programming Experience	15 years	15 years	18 years
Digital photo experience	2 years	2 years	4 years
PhotoFinder Experience	None	None	None
Frequently used Photo Browser	ACDSee and Canon photo	Irfanview	ACDSee
Types of personal media data stored in personal computer	Movies, Music, Photos, Home videos	Movies, Photos, Music	Movies, Comics, Video (e.g. Documentary, TV show, animation, etc.), Photos, Music, EBooks
Home Internet Access	DSL (768 Kbps)	Cable Model (1.5 Mbps)	Cable Modem (1.5 Mbps)

Table 6.2.1 Background information of 3 experienced subjects for the second user study. The users' background information for two novice users was omitted because it was already done in the first user study (See **Table 6.1.1**)

Practice Tasks:

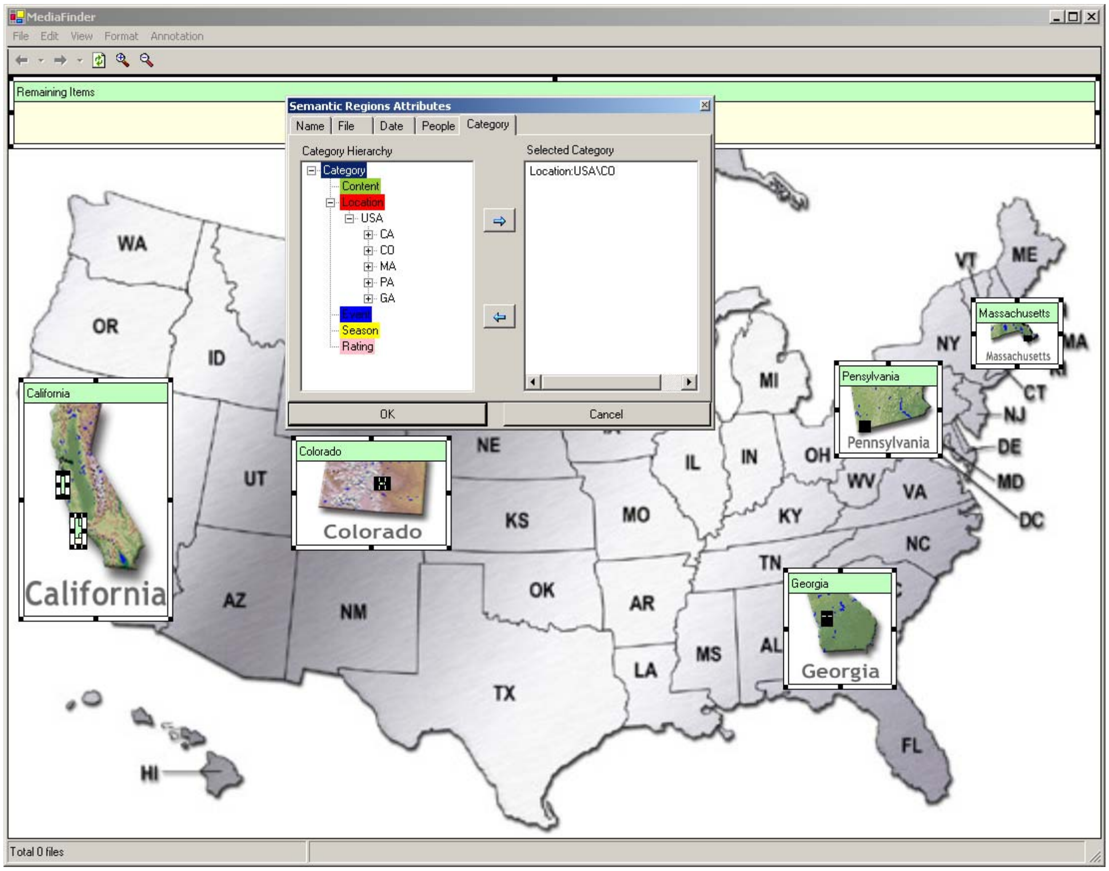
- Create a rectangular region on the MediaFinder interface.
- Resize and relocate the region as desired.
- Create two child regions inside a region.
- Load the background images of each region.
- Select two regions and delete them.
- Specify semantics to the regions. Then, modify or remove the semantics from them.
- Save the model you've created as a file.
- Load the model you've saved.
- Combine (import a model) the two models you've created.

After the completion of the practice tasks, they were given 4 tasks together with test data set. The test data set was the same as what was used for the first user study (281 Photos from CHI conference). In addition to photos, one simplified "Location" Semantic Regions template was given as an example(**Figure 6.2.1**).

Real Tasks:

- Make a "Location" model identical to the given example;

  1. Make the same model as the sample location model (**Figure 6.2.1**) you are given. Please confirm that your model works exactly the same as the given model by using the *fling-and-flock* metaphor.

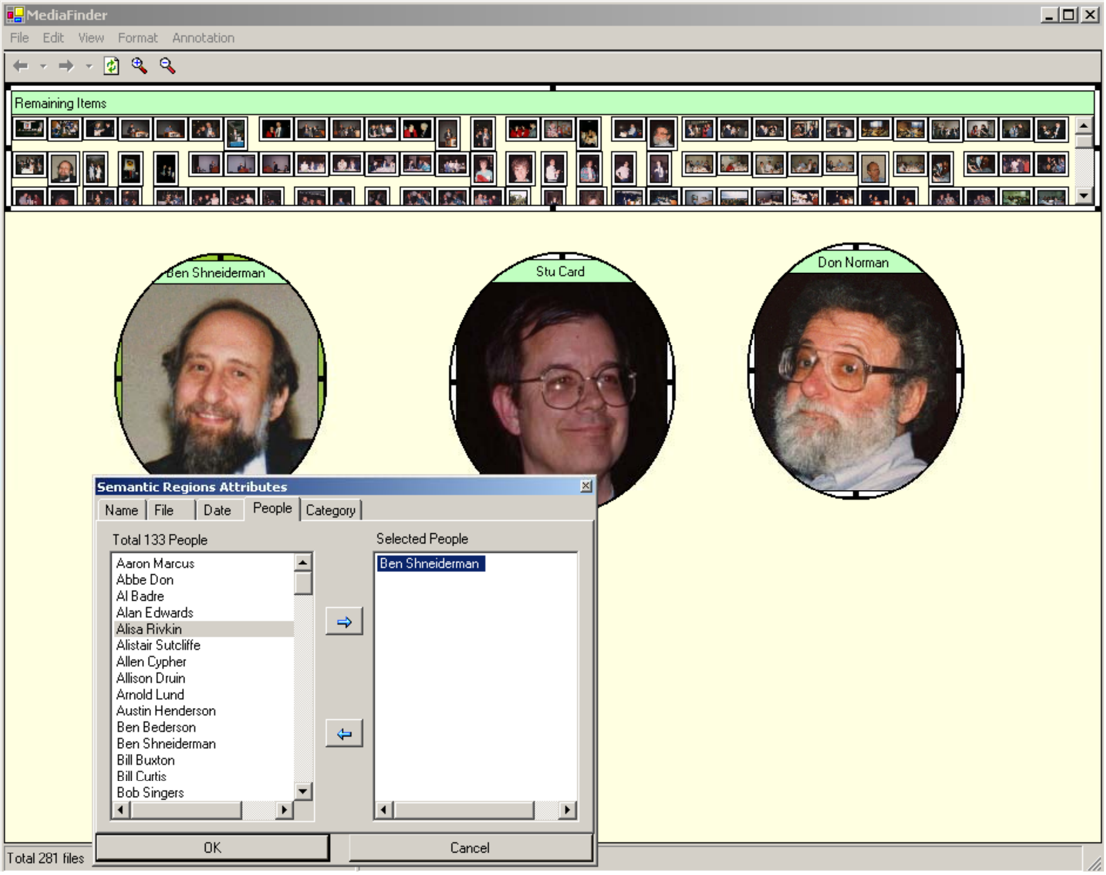


Selected Category
Location:USA\CO

Figure 6.2.1 US map mental model with 5 state regions and 5 child regions of cities. Users can specify location semantics for a region by selecting a tag from the location category in the Semantics dialogbox

- Make a "People" model without example;

2. Choose any three people from the people list and construct regions for them. (Load their pictures as the background images of the created regions) And see how many pictures were taken for each person between 1995 and 2000. (Specify Conjunctive Semantics)



Selected People: Ben Shneiderman

Figure 6.2.2 A people model consists of three Semantic Regions. Users can choose a person from the list of people in the semantic dialog box. The list of people was collected from the metadata of bound photos.

- Combine models

3. Combine two models that you have created and see what states each person has been to, and find out the states that all three people have been to. (Combine the models horizontally)

   How many pictures were taken for each person in California ?

   How many pictures of Shneiderman were taken in each state ?

   (Combine the models hierarchically)

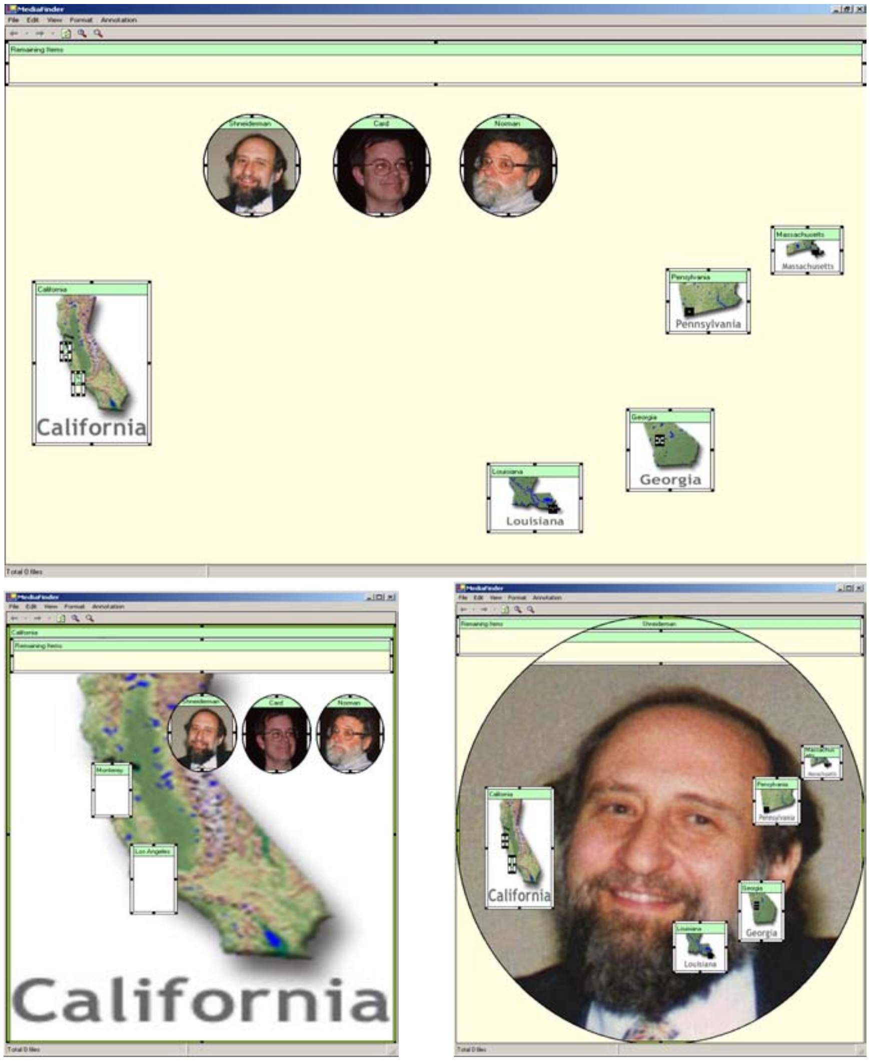


Figure 6.2.3 (a) People and location model combined horizontally (b) People model imported inside Location model (c) Location model imported inside People model

■ Free Construction

4. Make your own model with the given metadata such as date, people, location, name of conference, etc., and figure out any interesting facts from the created model.

   Example models; Seasonal calendar, yearly calendar, or monthly calendar.

To determine the degree of difficulty in constructing Semantic Regions, users' learning time, success rate, time to completion, cognitive trouble spots, and problems in MediaFinder interface were measured.

Finally, subjects were given the opportunity to freely explore the system, describe the problems with the MediaFinder interface, and offer suggestions for improvement through the post verbal survey (Appendix B.3).

6.2.2 Background Information

From the background survey, three of the subjects were determined to be expert personal computer users in terms of computer usage experience, knowledge about windows file system, object-oriented drawing tool and programming experience, but had experience with only basic visualization tools such as MS Excel charts. They handle various types of personal media such as video clips, audio clips, photos, books, etc. on their personal computers. They usually store their data in secondary storage such as hard disks, CDs, DVDs, etc. by defining their own categories and creating the corresponding folder hierarchies for later use. Although the entire folder structure was created by themselves, they said that it was time-consuming and difficult to find what they wanted from their storage because they mostly counted upon their long-term memories in searching the files but their folder structures were often poorly-organized, inconsistent, and frequently changed.

Two of the subjects had never made any annotations for their media except the name of the folder. Although the folder name was the one and only self-annotated metadata they had, they did not even have any consistent naming mechanism (sometimes they use the date and place as folder names, and sometimes they make use of event name, title of media, author name, genre, or diverse and mixed categories of the media). One subject, who had a large amount of media, said that he had to use an Excel file to maintain the media metadata and use it for search because he could not count upon his memory to manage thousands of offline files (CDs and DVDs). He also mentioned about a cataloging application called "whereisit (http://www.whereisit-soft.com)", which is designed to help users maintain and organize a catalog of their computer media collection, including CD-ROMs, audio CDs, diskettes, removable drives, hard drives, network drives, DVDs, or any other media that Windows can access as a drive. The basic goal for this application is to provide access to the contents of any media that users have from a cataloged database, even if the media itself is not available on the system. Also users can browse lists of files and folders, search by any criteria, use descriptions, thumbnails, categories, flags, etc.

Another subject said that he was using an MP3 management software called "mp3-explorer (http://www.mp3-explorer.com/explorer.htm)" because he often had a problem in remembering and searching music files in his computer. This software organizes music files automatically by artist, album, or genre (those are recorded in the mp3 file header) independent of the file folder hierarchy structure. Even though the recorded metadata was neither complete nor consistent, he said it often helped him to find the files he wanted and to see an overview of what he had in his storage.

6.2.3 Results

After a 10-minute demonstration about constructing Semantic Regions, the subjects seemed to understand most of the features and were able to complete the practice tasks with occasional help. In general, the subjects did not seem to have any major problem in learning the concepts and methods for constructing *Semantic Regions*. For the real tasks, all the subjects were able to complete all 4 questions. During the tests, subject 1 (one of novice users) asked for help in remembering how to import a model in task 3. It took 3 ~ 14 minutes for task 1, 3 ~ 6 minutes for task 2, 5 ~ 12 minutes for task 3, and 3 ~ 10 minutes for task 4 (see **Table 6.2.2**).

	Subject 1	**Subject 2**	**Subject 3**	**Subject 4**	**Subject 5**
Task 1	S (2 m 52 s)	S (14 m 57 s)	S (6 m 41 s)	S (9 m 1 s)	S (7 m 56s)
Task 2	S (3 m 23 s)	S (6 m 4 s)	S (3 m 35 s)	S (4 m 10 s)	S (5 m 35 s)
Task 3	SH (9 m 19 s)	S (12 m 10 s)	S (7 m 15 s)	S (5 m 30 s)	S (5 m 30 s)
Task 4	S (4 m 35 s)	S (3 m 54 s)	S (8 m 34 s)	S (10 m 3 s)	S (3 m 51 s)

Table 6.2.2 Success or failure of task (S: success, F: fail, SH: success with help) and time to complete (m: minute, s:second)

In task 1, all subjects were able to create regions on the MediaFinder main panel and specify the semantics according to the given template, even though they all accomplished it in different ways. No universal procedure could be found in users' strategies for creating Semantic Regions. For example, one subject accomplished the task according to the following steps: (1) Create five regions one by one from left to right (2) Change the titles of regions from left to right (3) Load a background image for each region from left to right (4) Create a child region inside a parent region (5) Load a background image for the child region (6) Specify the semantics to the child region (7) Specify the semantics to the parent region (8) Repeat steps (4) ~ (7) four times from the leftmost region to the rightmost region. On the other hand, another subject followed a different procedure: (1) Create a region (2) Load the background image for the region (3) Specify the semantics for the region (4) Create a child region inside the region (5) Load the background image for the child region (6) Specify the semantics to the child region (7) Repeat steps (1) ~ (6) four times to create five US state regions from left to right. Two of the subjects did not use the repetitive procedure partly because they felt it was tedious and partly because they forgot what the previous procedure was. Although all five subjects accomplished the task in five different ways, it was interesting to observe that they all created five regions and specified semantics with the same order from left to right.

No serious cognitive trouble spots were found in task 1, but a few comments concerning the MediaFinder interface came out from the subjects. First of all, most subjects did not like using the middle mouse button to create a region because they thought it was hidden and not intuitive. They preferred switching between the creation and the selection mode and using the left mouse button for the click-and-drag metaphor as in most object-oriented drawing tools.

Some of the subjects wanted a way to overview or preview the region hierarchy without maximizing the region. For instance, how many child regions or grand-child regions do exist in a region, how does their spatial arrangement look, what is the maximum depth of the hierarchy of a region, and so on.

Three of the subjects mentioned that it would be nice if they could create child regions without maximizing the parent region. They reported that it took too many steps to create child regions, specify the semantics for them and delete them because of the steps needed for maximizing and minimizing the parent region.

Finally, two subjects mentioned that they were a bit confused about switching between the several views of a region (list view, large icon, small icon, thumbnails, details, intersection, number, and background) and they often forgot which view they were looking at especially when a region was maximized or minimized. Also, they said that they thought they could create child regions in any view, but, in fact, they were not allowed to create child regions in other views except the background view.

In task 2, subjects had no problem in creating regions and specifying the semantics for them because they had already accomplished a similar task in task 1. Two major points were observed carefully in task 2. The first one was about whether users could understand and specify the conjunctive semantics. The second one was about whether a region should support more complex semantics (e.g. disjunctive and complement) other than conjunctive semantics.

In summary, subjects were able to understand the conjunctive semantics even though they did not know what the conjunctive query meant. They did not feel it unusual to narrow down the search space by accumulating the semantics in different categories, as in the case of Ben Shneiderman's (people category) photos taken in Maryland (location category) between 1996 and 2000 (date category). Also it was interesting to observe that all subjects took it for granted that all the selected categories would be joined conjunctively. All of the subjects were also able to add a new semantic feature to a region, change the semantics of a region, and remove the semantics from a region without any trouble.

Right after task 2 was done, subjects were asked whether they thought it would be necessary to support more complicated semantics for a region. Four of the subjects mentioned that the conjunctive semantics would be enough because complex queries could be formed by applying Boolean operations among the regions. They agreed that the semantics of a region should be as simple as possible to minimize their efforts to remember what each region meant. In addition, they said that if they were given to use MediaFinder in the future, they would create regions so as to have only one or two conjunctively joined semantics. On the contrary, one subject strongly insisted that the more complicated semantics should be supported in MediaFinder because it would enable users to avoid producing unnecessary regions. He presented an example of creating a region that represented a group of people in a department of a certain company. In this example, he said that he did not want to create 30 separate regions to represent each person in his department and then apply a Boolean OR operation to the 30 regions. Rather, he would like to specify disjunctive semantics to a region so that it could accommodate 30 people at once with one simple step. As shown in this example, there should be more research on the relationship between the conceptual size of a region and the complexity of the semantics for a region.

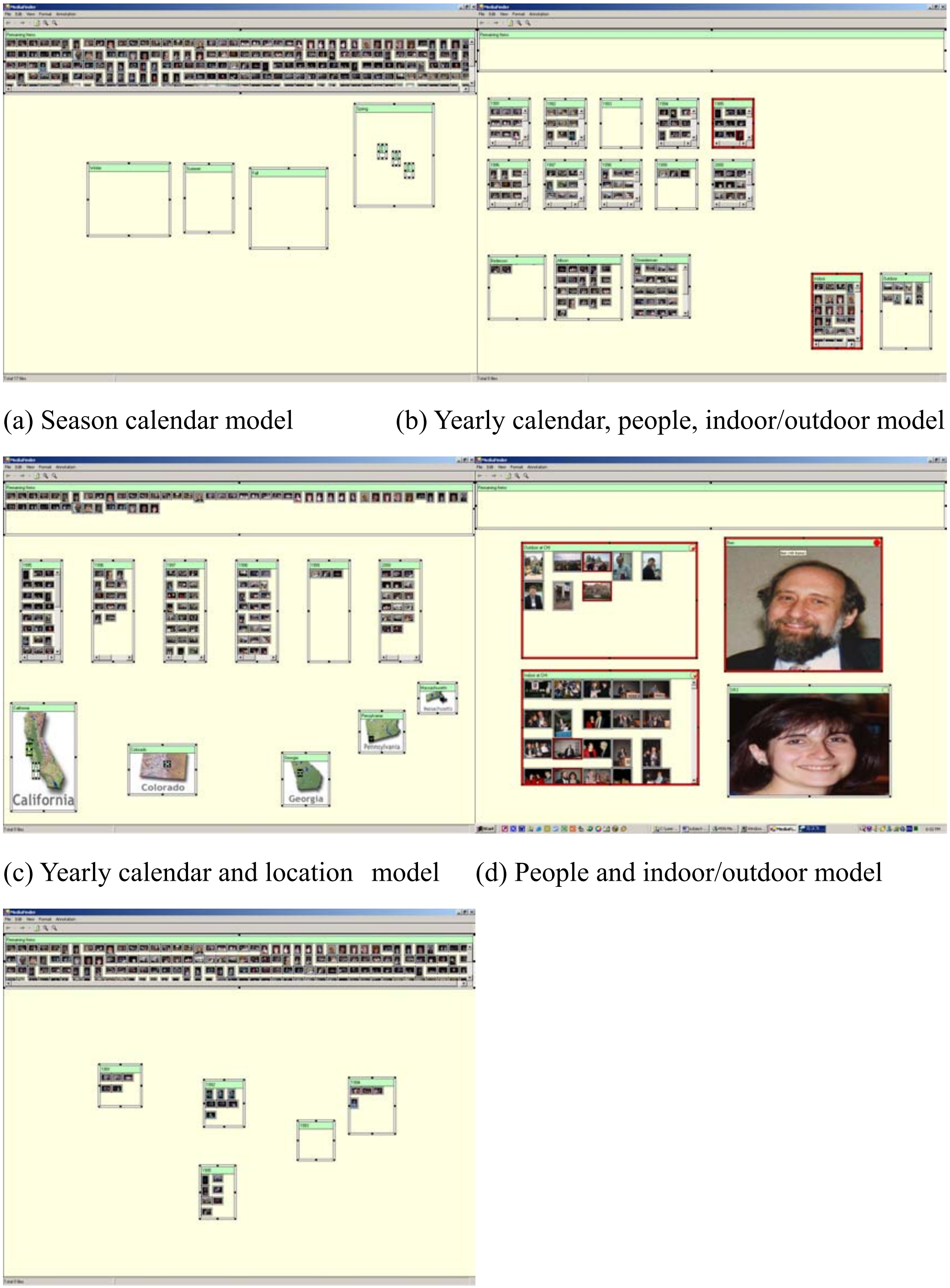
There were two more comments concerning the MediaFinder interface in task 2. Two of the subjects mentioned that it would be nice if they could specify a semantic of multiple regions at once. They reported that it was boring to specify the same date range (1995~2000) semantics to the multiple regions one at a time. Another subject suggested that it would be great if there were a way to create multiple regions simultaneously while automatically selecting the appropriate predefined semantics, rather than the standard two-step procedure (create a region first and then specify the semantics). For example, users could create 10 people regions simultaneously just by dragging the names of people from the semantic dialog box onto the MediaFinder panel instead of creating 10 regions individually and specifying the names of people for each region one by one.

In task 3, two of the subjects tried to solve the problem by applying Boolean operations and changing the conjunctive semantics of the regions repeatedly. During the task, they were reminded that they could combine two models using the “import” menu. Four of the subjects remembered how to import a model and combine it with the existing model, but one subject did not remember and asked for help. Some subjects confounded the “open” menu with the “import” menu during the task and thus they replaced the current model with a newly opened model instead of combining them. However, they quickly corrected their mistakes by importing the old model again. An interesting fact was found during the test. That is, subjects understood the vertically combined model more easily than the horizontally combined model. In other words, subjects were better at understanding the hierarchically combined model such as a location model inside the people model or a people model inside the location model than understanding the laterally combined model in the same level such as a people model together with a location model. Furthermore, subjects said that they were less confused when the heterogeneous models (models that have different categories) were combined hierarchically than when the heterogeneous models were combined laterally. Although no significant performance differences were found in this user study, further study on this issue might be interesting.

In task 4, subjects were asked to freely construct their own models if they had any in their minds, or they were asked to create a calendar model. Most of the subjects spent one or two minutes to think about what would be an interesting model. Four of the subjects constructed time-related models and three of the subjects constructed multiple models and combined them. The first subject created a season calendar model that consisted of spring, summer, fall, and winter regions (**Figure 6.2.4**(a)). In addition, the subject created three child regions inside the spring region to represent three months in spring. However, the subject created the one-year period of a seasonal calendar because the subject did not know how to specify the date range of multiple years for each season. This was another demand for the complex semantics for a region. The second subject constructed three different models (yearly calendar, people, and indoor/outdoor photos) and combined them (**Figure 6.2.4**(b)). With the combined model, the subject tried to figure out who had the outdoor pictures and when those pictures were taken. The third subject created a yearly calendar from 1995 to 2000 and combined it with an existing location model that was used for task 1 (**Figure 6.2.4**(c)). With this model, the subject easily figured out the time of the photos based on the location and vice versa. The fourth subject made use of *Semantic Regions* for formulating queries. The subject tried to find out who had the most outdoor pictures. So the subject constructed two regions for indoor and outdoor pictures and constructed a few regions of people whom the subject knew. Then, the subject found out who had the outdoor pictures by placing the mouse cursor over the outdoor picture region and observing the highlighted people regions. Then the subject applied the Boolean AND operation between the outdoor picture region and the entire highlighted people regions one by one. The fifth subject created a yearly calendar of the Ben Shneiderman model by specifying conjunctive semantics to the regions.

Because the photos used for the user study had limited metadata (they only had location, people, date, conference name, and indoor/outdoor attributes) and the subjects were not familiar with the people in the photos, there was a clear limitation for the subjects to create some appealing models in a short time. However, it was quite impressive to see that both novice and expert users could create their own models with MediaFinder after just about 30 minutes of use. In addition, it was quite interesting to observe that no subject created a region that had more than two conjunctive semantics. In summary, the subjects seemed to consider a region as a basic organizing unit and wanted to make it as simple as possible.

Three questions were frequently asked during task 4. The most frequently asked question was how to construct multiple regions at once and specify the repetitive semantics for the regions with minimum effort. The subjects seemed to easily get tired of constructing repetitive regions such as the monthly calendar or yearly calendar. They also wanted some easier ways to specify more complicated semantics for multiple regions simultaneously, for instance, defining seasons to the regions. The improvement of the MediaFinder user interface concerning this issue is described in Section 4.5.2.



(a) Season calendar model (b) Yearly calendar, people, indoor/outdoor model

(c) Yearly calendar and location model (d) People and indoor/outdoor model

(e) Yearly calendar model with conjunctive semantics

Figure 6.2.4 Screenshots of the models created by the subjects in task 4

The second question was how they could easily arrange the multiple regions on the 2D plane as they wished. They often tried to format regions to have the same size, to have the same shape and to be aligned and spaced horizontally or vertically for clearer and better representation of their mental models. Supporting rich formatting features may help users construct their mental models more easily and clearly. In addition, studying the differences of formatting features between MediaFinder (mental model building tool) and object-oriented drawing tools would be an interesting topic for future research.

The last question was about how they could know what kinds of semantics were available in specifying the semantics for a region. Most subjects wanted to know what kind of attributes each picture had as well as how many pictures were annotated with each attribute before they constructed their mental models. The query preview of the semantics for bound media might work for this problem.

In the verbal post survey, most subjects showed their interests in MediaFinder. As to the subjects' general reaction to MediaFinder, they clearly showed enthusiasm. However, there may have been social pressure to respond positively, since the subjects knew that the administrator of the experiment was also the developer of MediaFinder.

For the questionnaire about ease of use (scale 1 to 9) for MediaFinder interface, they marked an average of 7.8 points (See **Table 6.2.3**), which was not much different but a little lower than the first user study (8.2 points). They reported that the MediaFinder interface was not particularly hard for them to use compared with other interfaces and the file explorer, but two of the subjects mentioned that managing many regions on a MediaFinder interface was a little bit annoying just as it was to manage many independent windows over a computer screen.

	Subject1	Subject2	Subject3	Subject4	Subject5
Ease of use (1: difficult, 9:easy)	8	7	8	8	8
Willingness to use with given metadata (1: not interested, 9:willing to use)	8	8	9	9	7
Willingness to use with manual annotation (1: not interested, 9:willing to use)	5	6	4	3	8
Helpfulness of MediaFinder Interface (1: disturbs, 9:helps)	7	8	9	8	8
Preference of MediaFinder to other tools (1: not at all, 9: certainly)	5	7	8	8	7

Table 6.2.3 User satisfaction survey on MediaFinder (1 – 9 scale)

For the questionnaire about the willingness to use (scale 1 to 9), they gave an average of 8.2 points, which was also a little bit lower than the first user study (8.8 points), under the condition that the metadata was given beforehand or automatically generated. Just as in the first user study, subjects were very concerned about the annotation process. They mentioned that they were not so much interested in using MediaFinder unless the metadata is automatically extracted or collected from the photos and they marked an average of 5.2 points, which was a little bit higher than the first user study, for “willingness to use with manual annotation.” For the questionnaire about the helpfulness of the MediaFinder interface, they all gave high points (an average of 8 points). However, the average points of the questionnaire about the preference of MediaFinder over other tools was lower (an average of 7 points) than those of previous questionnaires because some of the subjects did not think that they needed a powerful tool for personal media management for the moment.

After the test, the PhotoFinder4 beta version was shown to everybody especially from the standpoint of creating a user-defined category tag, bulk annotation, and direct annotation. All of the subjects except one reported that they would like to use MediaFinder if it had the same annotation features. However, one subject (who had a large amount of data) was still skeptical of using a manual annotation tool for managing his personal media data.

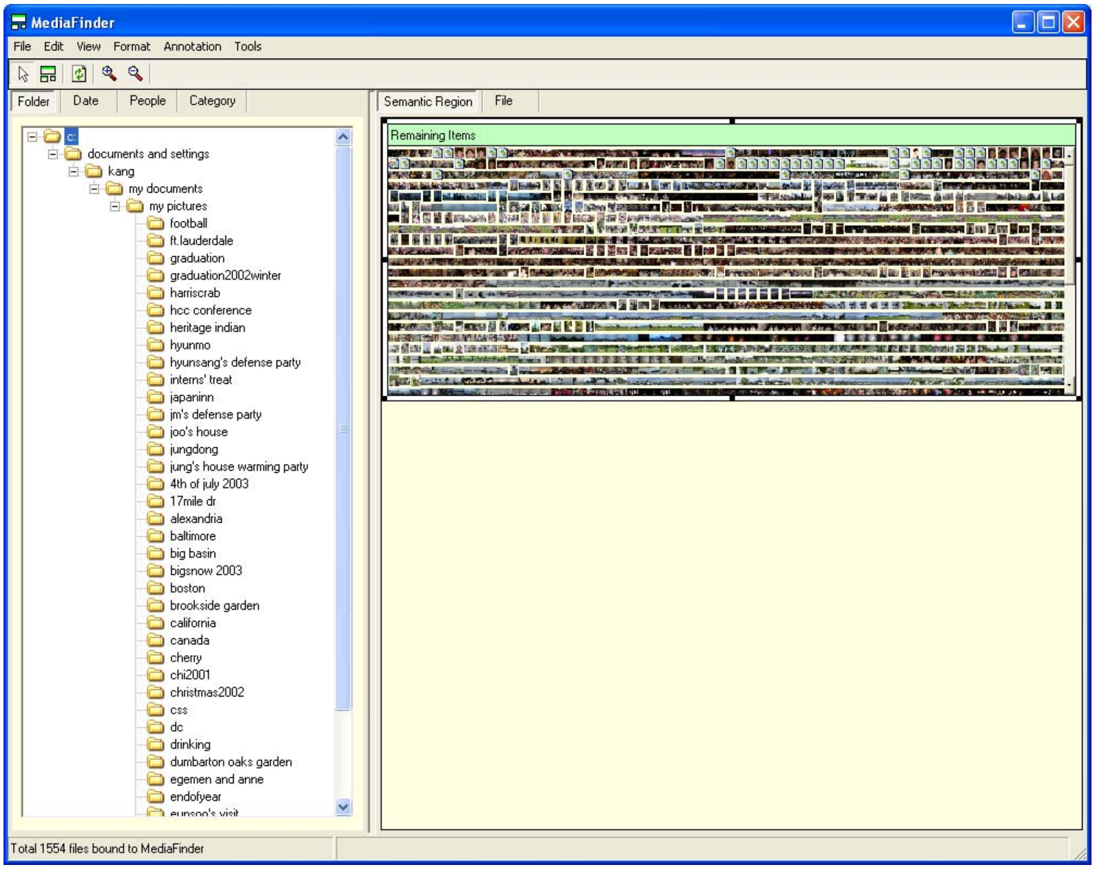
After the study, the subjects were asked to freely comment on the MediaFinder and the concept of Semantic Regions. One subject mentioned that he liked the idea of spatial organization and dynamic grouping very much but he was not so much interested in constructing his own models using MediaFinder because it took too much time and effort. The subject suggested that it could be better to provide various built-in templates rather than let users create their own, or add more automatic features to reduce the steps for region construction and semantic specification. For instance, just dragging the people's names from the semantic dialogbox onto the MediaFinder creates the people regions or dragging the name of the month or year creates regions with the corresponding date range semantics. Two of the subjects suggested a feature for grouping *Semantic Regions*, which could mediate between the inter-region Boolean operations and intra-region semantic specification.

6.3 Enhancement

The results of the user studies demonstrate that users, with some training, are able to construct their own mental models for personal media management with MediaFinder. The studies also helped to identify improvements to the user interface that could dramatically reduce the need for training, improve user performance, and decrease error rates. These enhancements focus on reducing the need for semantic specification and providing a control panel for semantics collected from the personal media items bound to MediaFinder.

6.3.1 Semantics Preview Control Panel

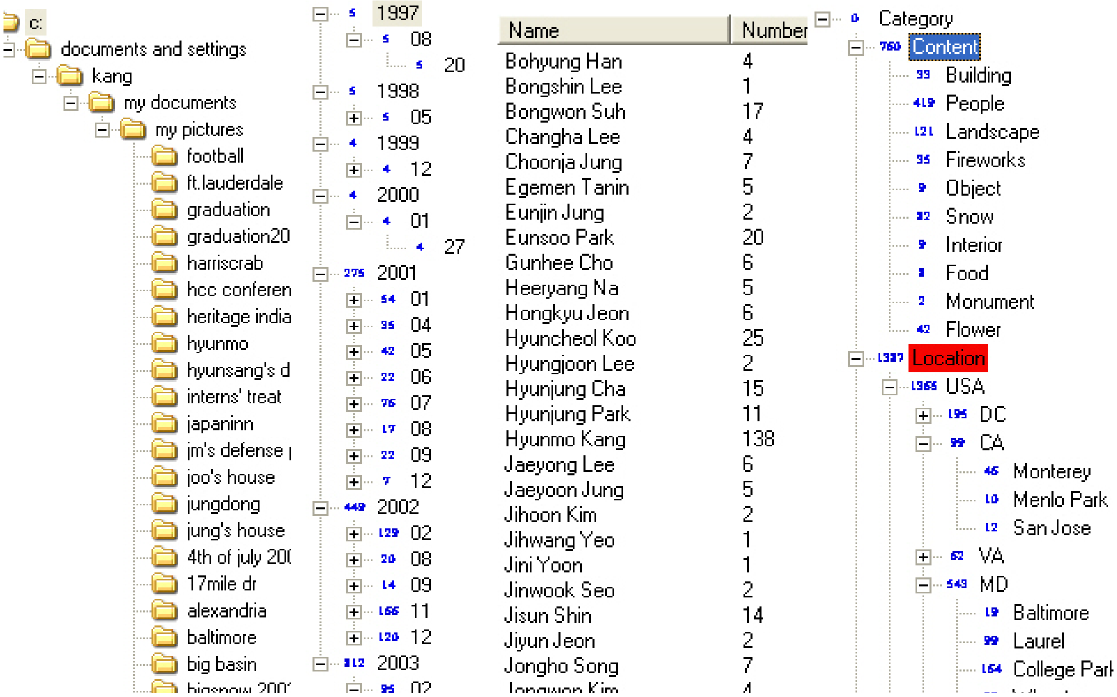
The MediaFinder user interface has been redesigned similarly to the Microsoft file explorer so that users can feel familiar with the interface. The overall MediaFinder user interface is composed of two separate windows: the Semantics Preview Control Panel and the main workspace panel (**Figure 6.3.1**). The Semantics Preview control panel (left window) is designed to show all of the metadata collected from the personal media items bound to MediaFinder. The collected metadata is reorganized based on four different hierarchies: Folder, Date, People and Category. With the multiple hierarchy views, users can see the different perspectives of the bound personal media items. The metadata collected in the Semantics Preview Control Panel is not only used as a query preview of the metadata but also used in specifying the semantics for a region and annotating the personal media items. In addition, they can also be used for *Semantic Regions* construction, which will be described in the following section. On the other hand, the main panel is used for visualizing the actual personal media items either in spatially organized form or in linear listed form.



Total 1554 files bound to MediaFinder

Figure 6.3.1 The MediaFinder interface consists of two separate windows: the Semantics Preview Control Panel (left window) and the Main Panel (right window). The Semantics Preview Control Panel is composed of four tab windows to show all the metadata collected from the bound personal media items. The Main Panel is composed of two tab windows to visualize the personal media items either in spatially organized form or in linear listed form.

The Semantics Preview Control Panel is composed of four tab windows to represent the different hierarchy of the metadata (**Figure 6.3.2**). The Folder view represents the actual folder structure of the personal media items residing in the file system (**Figure 6.3.2**(a)). The Date view shows all the file creation date information of the personal media items. The numbers shown in front of each year, month, and day hierarchy represents the total number of the personal media items that fall under the category (**Figure 6.3.2**(b)). The People view shows the people metadata annotated in the personal media items (**Figure 6.3.2**(c)). It also shows the total number of the personal media items that contain the corresponding people annotation in the “number” column. The Category view shows the user-defined category metadata (e.g. location, content, rating and so on) recorded in the personal media items (**Figure 6.3.2**(d)).



- c:
  - documents and settings
    - kang
      - my documents
        - my pictures
          - football
          - ft.lauderdale
          - graduation
          - graduation20
          - harriscrab
          - hcc conferen
          - heritage india
          - hyunmo
          - hyunsang's d
          - interns' treat
          - japaninn
          - jm's defense |
          - joo's house
          - jungdong
          - jung's house
          - 4th of july 200
          - 17mile dr
          - alexandria
          - baltimore
          - big basin
          - bigsnow 200

(a) Folder view

- 5 1997
  - 5 08
    - 5 20
- 5 1998
  - 5 05
- 4 1999
  - 4 12
- 4 2000
  - 4 01
    - 4 27
- 275 2001
  - 54 01
  - 35 04
  - 42 05
  - 22 06
  - 76 07
  - 17 08
  - 22 09
  - 7 12
- 449 2002
  - 129 02
  - 20 08
  - 14 09
  - 166 11
  - 120 12
- 312 2003
  - 95 02

(b) Date view

Name	Number
Bohyung Han	4
Bongshin Lee	1
Bongwon Suh	17
Changha Lee	4
Choonja Jung	7
Egemen Tanin	5
Eunjin Jung	2
Eunsoo Park	20
Gunhee Cho	6
Heeryang Na	5
Hongkyu Jeon	6
Hyuncheol Koo	25
Hyungjoon Lee	2
Hyunjung Cha	15
Hyunjung Park	11
Hyunmo Kang	138
Jaeyong Lee	6
Jaeyoon Jung	5
Jihoon Kim	2
Jihwang Yeo	1
Jini Yoon	1
Jinwook Seo	2
Jisun Shin	14
Jiyun Jeon	2
Jongho Song	7
Jongwon Kim	4

(c) People view

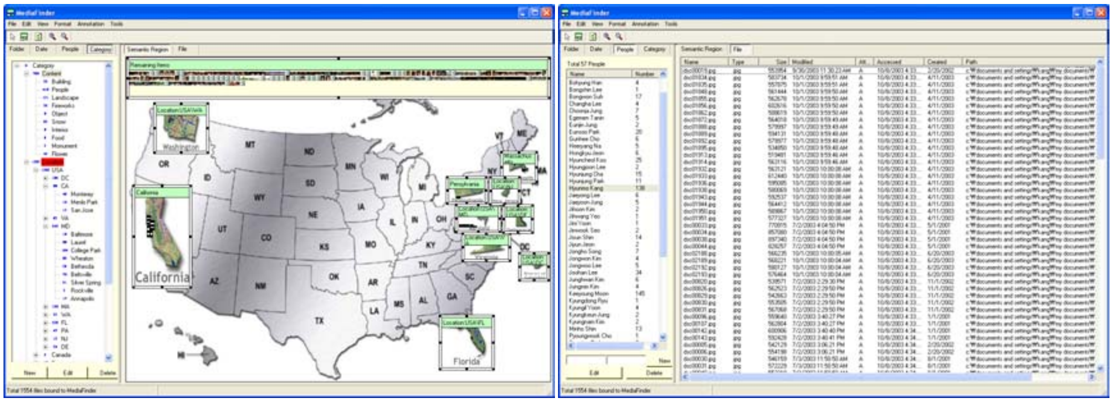
- 0 Category
  - 760 Content
    - 33 Building
    - 419 People
    - 121 Landscape
    - 35 Fireworks
    - 9 Object
    - 32 Snow
    - 9 Interior
    - 3 Food
    - 2 Monument
    - 42 Flower
  - 1337 Location
    - 1365 USA
      - 195 DC
      - 99 CA
        - 46 Monterey
        - 10 Menlo Park
        - 12 San Jose
      - 62 VA
      - 543 MD
        - 19 Baltimore
        - 99 Laurel
        - 164 College Park

(d) Category view

Figure 6.3.2 Four different views of the collected metadata shown in the Semantics Preview Control Panel.

The main panel is composed of two tab windows: the Semantic Regions panel and the File List view panel. The Semantic Regions panel is used for constructing the *Semantic Regions* and using them to explore the personal media items that are bound to MediaFinder (**Figure 6.3.3**(a)). On the other hand, the File List view panel shows the personal media items in the form of a linear list of file items just as in the file explorer (**Figure 6.3.3**(b)). Once metadata is selected in the Semantics Preview

Control Panel, all the personal media items containing the metadata are listed in the file list view panel.

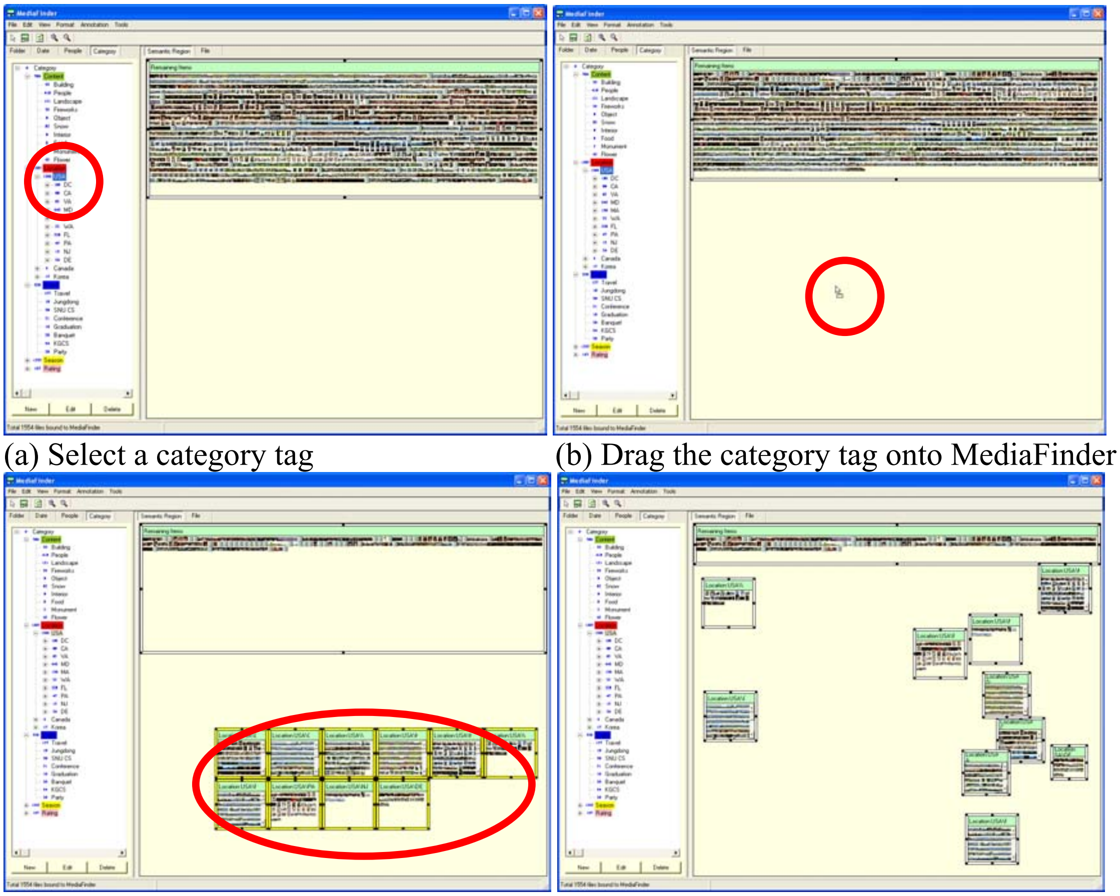

MediaFinder
File Edit View Format Annotation Tools
Folder Date People Category
Semantic Region File
Remaining Items
Location:USA\WA
Washington
California
Florida
New Edit Delete
Total 1554 files bound to MediaFinder
Total 57 People
Name Number
Name Type Size Modified Att... Accessed Created Path

(a) Semantic Regions panel (b) File List view panel

Figure 6.3.3 Two different views of the main panel:

6.3.2 Enhancement of Semantic Regions Construction

The study on construction revealed that the two step approach of creating new regions and then specifying semantics for each region was the primary difficulty for users because it took too much time and effort. Reducing the need to specify semantics in this manner would be a major benefit. While the capability for manually specifying semantics for a region empowers the expressiveness of semantics, shortcuts are possible for common simple situations. An applicable HCI design principle is: make the common task easy, possibly at the expense of making rare tasks more difficult. The MediaFinder user interface has been redesigned to support a simple drag-and-drop *Semantic Regions* construction to reduce the steps for region construction and semantics specification.



(a) Select a category tag (b) Drag the category tag onto MediaFinder

(c) The child category regions are created automatically with the corresponding semantics (d) Arrange the regions spatially

Figure 6.3.4 Simplified region construction and semantics specification with one simple drag and drop action

Figure 6.3.4 shows the way to construct Semantic Regions and specify semantics for each region with just one simple drag-and-drop action. Users can choose a category tab and select a category tag from the semantic control panel (**Figure 6.3.4**(a)). The selected category tag can be dragged onto the MediaFinder main panel as in **Figure 6.3.4**(b). When the dragged tag is dropped on the main panel, MediaFinder automatically creates the same number of *Semantic Regions* as the number of child categories below the selected category tag in the category hierarchy (**Figure 6.3.4**(c)). Simultaneously, each region is specified with the child category values. Then, all the personal media items bound to MediaFinder are dynamically regrouped according to the semantics specified in the newly created regions. Users can spatially rearrange the newly created regions based on the specific tasks or their mental models (**Figure 6.3.4**(d)).

6.4 Summary

Overall, all the subjects, even if they were novice users, succeeded in grasping the cognitive concept of Semantic Regions and using the MediaFinder interface. Users liked the concepts of spatial organization and dynamic grouping of personal media based on their mental models. Users successfully and enthusiastically designed and constructed their own mental models using MediaFinder. They showed creativity and variation in their designs. These users seemed to be ready for more advanced tools than simple browsers for personal media management and exploration.

For practitioners, these studies indicate that Semantic Regions can be used in its present form, or that the concept of *Semantic Regions* can be implemented into other systems, to greatly enhance the user experience.

For researchers, several open questions require further study. The conceptual size of semantic regions needs to be empirically estimated. Also, supporting more flexible and powerful semantics without increasing the complexity of user interface needs to be examined. Various interface designs to reduce the steps needed for Semantic Region construction should also be explored. Finally, additional evaluation will be needed to examine the effects of MediaFinder interface improvements identified in the study on Semantic Region construction.

Chapter 7:

Conclusion

Semantic Regions is a conceptual model, user interface, software architecture, and implemented system that allows users to rapidly explore and efficiently manage personal media items stored on their personal computers. Users can dynamically construct a variety of their mental models on the fly and apply them to personal media items through the *fling-and-flock* metaphor. The *Semantic Regions* are spatially arranged on 2D space and used for various personal media management tasks such as organization, meaning extraction, search, navigation, indexing, and distribution.

Usability studies on MediaFinder revealed benefits, cognitive issues, and usability concerns. Both novice and experienced users succeeded in grasping the cognitive concept of *Semantic Regions* and using the MediaFinder interface. Users have a preference for the concepts of spatial organization of information and dynamic regrouping of personal media items based on their mental models and specific tasks.

7.1 Contributions

This research on *Semantic Regions* contributes five major innovations:

- **Conceptual model:** *Semantic Regions* provides a formal model of spatial organization and dynamic reorganization of personal media data based on users' mental models. It extends the concept from the current system-oriented file management system to a user-oriented personal media management system by employing the semantics of personal media data. In addition, an interactive metaphor, *fling-and-flock*, that is designed and implemented for binding personal media items to a variety of users' mental models, has applicability for visual organization of numerous data domains including scientific and statistical data.

- **User interface:** The design of a working environment for constructing and operating *Semantic Regions* is another major contribution. The ways of specifying the semantics for a region, showing the distribution of metadata collected from the personal media items, indexing personal media items, applying Boolean operations for search, navigating the hierarchy of *Semantic Regions* were the main design issues. The visual design, dynamic aspects, and user controls implemented in *Semantic Regions* are also among its many contributions to user-interface design.
- **Implementation of MediaFinder:** MediaFinder uses *Semantic Regions*, the *fling-and-flock* metaphor, and flexible metadata categorization capabilities to support exploration and management of personal media data sets. Implemented in Visual Basic .Net, MediaFinder as well as *Semantic Regions* uses object-oriented design techniques that support the use of subclassing to easily add new classes of features.
- **Evaluation of *Semantic Regions*:** Although more work remains to be done in empirically characterizing the strengths of *Semantic Regions* as a personal media management mechanism, studies conducted so far have led to an increased understanding of the strengths of *Semantic Regions*. Further studies

will attempt to refine this understanding, with the ultimate goal of generalizing results to apply to other 2D user interface widgets.

- **Framework for extending the query model:** *Semantic Regions* is just a starting point. There are a number of possible extensions to *Semantic Regions* that might be used to increase the expressive power of the query language. Further work in this area will be needed to identify the potential extensions that are interesting and relevant to user tasks as well as realistically achievable.

7.2 Future Work

This work has led to a wide range of possibilities for future work. Extending the query language, implementing new classes of queries, further empirical evaluations, extension to other types of data, and examination of new algorithmic techniques are just a few of the challenging and interesting areas that may be suitable for closer examination.

7.2.1 Re-implementation of MediaFinder

As a research prototype, MediaFinder was implemented with a focus on demonstration of capabilities and exploration of ideas. A new implementation, based on lessons learned from the existing prototype, would simplify future work and lead to increased flexibility and extensibility.

Currently, MediaFinder assumes that the entire data set is in main memory and allows MS Windows operating system to handle this using virtual memory. This assumption limits the flexibility of MediaFinder to scale to a larger data set. Ideally, the data management code would be separated from other codes with clear abstraction that could support the possibility of importing alternative data storage or a database model.

7.2.2 Scaling

Personal media data stored on users' individual machines are very large, both in terms of the number of items and the total size of items. Currently, users have hundreds of thousands of personal media items on their storage, but data may grow exponentially in the near future due to the extraordinary progress of computer hardware and cyberization technologies. Scaling the *Semantic Regions* model to accommodate a large number (perhaps millions of items) would require improvements in search algorithm and the rendering portion of the system. Alternatively, a larger data set might be clustered into smaller sets of manageable size. Other possibilities include extending MediaFinder to use disk based indices and possibly query previews to query data that are too large to fit into main memory.

7.2.3 Further evaluation

Current evaluations of *Semantic Regions* and MediaFinder have only provided the results of two heuristic usability studies. The first usability study was to demonstrate whether users can understand and use *Semantic Regions* for managing personal media data and the second one was to demonstrate whether users can create *Semantic Regions* based on their mental models. These usability studies were mainly done by observations and interviews with test subjects.

In addition to the previous heuristic evaluations, some well-designed controlled experiments are needed to identify and measure the benefits of MediaFinder for realistic data and queries. These empirical studies might include an experiment to compare the three file management systems: a system-oriented file browser like MS windows explorer, a file browser using user-defined high level semantics with simple interface (e.g. semantic file system), and MediaFinder in order to evaluate the effectiveness of MediaFinder over other file management systems.

7.2.4 Extension to other types of data

Although current research only focuses on personal media data sets, generalizing the concept of *Semantic Regions* to apply more challenging data sets presents further opportunities for interesting work. *Semantic Regions* and MediaFinder were originally designed to support personal media data that have both file properties and user-defined metadata. However, a potentially interesting generalization of *Semantic Regions* and MediaFinder might involve support for both structured and unstructured data. An item can be a record in a relational database table and the attributes of the table can be used for the semantics of an item. With this extension, MediaFinder can deal with structured data such as Census data or genomic data to help users understand data more intuitively and construct queries more easily.

In the bigger picture, the concept of *Semantic Regions* could provide a solution to a rising new problem on the web. Larger databases are increasingly used on the web. With applications such as e-commerce and warehousing, more of the web is data driven and consequently XML is on the rise. However, user interface on the web is improving slowly. MediaFinder can be extended to accommodate the unstructured XML data, just as it handles the user-defined category of personal media data, to visualize or formulate queries over the web database.

7.3 Conclusions

I believe that the concept of *Semantic Regions* and the MediaFinder interface can help users explore their personal media data more rapidly and manage them more efficiently. MediaFinder users can construct the mental models they need for their data and personal media management tasks such as organization, meaning extraction, search, navigation, indexing, and distribution. Without MediaFinder, such tasks would be difficult and time-consuming to achieve.

However, this research is just the beginning. The concept of *Semantic Regions* opens new possibilities for applied information visualization. I believe it is one step towards helping a wider range of users explore data, make complex decisions and apply their creativity.

References

1. Adar, E., Kargar, D., Stein, L.A., Haystack: per-user information environments. *In Proceedings of the eighth international conference on Information and knowledge management*, 413 - 422, ACM Press, 1999.
2. Ahlberg, C., Shneiderman B., Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays*, Proceedings of the ACM CHI '94 Conference*, 313-317, 1994.
3. Ahlberg, C., Williamson, C., Shneiderman, B., Dynamic Queries for Information Exploration: An Implementation and Evaluation*, Proceedings of the ACM CHI '92 Conference*, 619-626, 1992.
4. Beaudouin-Lafon, M., Novel Interaction Techniques for Overlapping Windows, *Proceedings of UIST2001, ACM Symposium on User Interface Software and Technology*, 153 – 154, 2001
5. Bederson, B.B., Hollan, J. D., Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics, *Proceedings of UIST94, ACM Symposium on User Interface Software and Technology*, 17 - 26, 1994.
6. Bederson, B.B., PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps, *UIST 2001, ACM Symposium on User Interface Software and Technology, CHI Letters*, *3*(2), 71 – 80, 2001.
7. Bell, B., Feiner, S., Dynamic space management for user interfaces, *Proceedings of UIST2000, ACM Symposium on User Interface Software and Technology*, CHI Letters 2(2):239 - 248, 2000.
8. Bell, C. G., The Cyber All Project: A Personal Store for Everything, *Microsoft Research Technical Report MSR-2000-75*, July, 2000.
9. Bell, G., A Personal Digital Store, *Communications of the ACM, Vol. 44, No. 1* (January 2001), 86 - 91, 2001.
10. Bell, G., Gray, J., Digital Immortality, *Communications of the ACM, Vol. 44, No. 3* (March 2001), 28 – 30, 2001.
11. Burbeck, S., Applications Programming in Smalltalk-80 ™: How to use Model-View-Controller (MVC), http://st-www.cs.uiuc.edu/users/smarch/st-docs/mvc.html.

12.Calvary, G., Coutaz, J., Thevenin, D., A Unifying Reference Framework for the Development of Plastic User Interfaces, *EHCI'01, IFIP WG2.7 (13.2) Working Conference*, Toronto, 2001.

13.Card, S., Mackinlay, J., Shneiderman B., Readings in Information Visualization: Using Vision to Think*, Morgan Kaufmann*, 1999.

14.Chen, H., Dumais, S.T., Bringing order to the web: Automatically categorizing search results, *Proceedings of the ACM CHI'00 conference,* 145-152, 2000.

15.Chuah, M.C., Dynamic aggregation with circular visual designs, *Proceedings of the IEEE Symposium on Information Visualization*, 35 - 43, October, 1998.

16.Chui, M., Pattern, Procedurality & Pictures: Factors Affecting Boolean Query Interface Design for the Web, *Proceedings of ACM SIGIR99,* 1999.

17.Clarkson, B., Mase, K., Pentland, A., The Familiar: a living diary and companion, *Proceedings of the ACM CHI2001 extended abstracts*, 271-272, April, 2001.

18.Cockbrun A., Mckenzie B., Evaluating the Effectiveness of Spatial Memory in 2D and 3D Physical and Virtual Environments, *Proceedings of the ACM CHI2002 conference*, 203 - 210, 2002.

19.Derthick, M., Kolojejchick, J., Roth, S.F., An Interactive Visual Query Environment for Exploring Data, *Proceedings of the 10th annual ACM symposium on User interface software and technology*, 189-198, 1997.

20.Dourish, P., Edwards, W. K., LaMarca, A., Salisbury, M., Presto: An Experimental Architecture for Fluid Interactive Document Spaces, *ACM Transactions on Computer-Human Interaction, 6(2),* 1999.

21.Fertig, S., Freeman, E., Gelernter, D., Lifestreams: An Alternative to the Desktop Metaphor, *Proceedings of the ACM CHI96 Conference*, New York, 410 – 411, 1996.

22.Fishkin, K., Stone, M.C., Enhanced Dynamic Queries via Movable Filters, *Proceedings of the ACM CHI '95 Conference*, 415-420, 1995.

23.Furnas, G. W., Generalized Fisheye Views, *Proceedings of the 1986 Conference on Human Factors in Computer Systems,* 16 - 23, April, 1986.

24.Gifford, D.K., Jouvelot, P., Sheldon, M.A., O'Toole, Jr., J.W., Semantic File Systems, *13th ACM Symposium on Operating Systems Principles*, October, 1991.

25.Goldstein, J., Roth, S. F., Using Aggregation and Dynamic Queries for Exploring Large Data Sets, *Proceedings of the ACM CHI '94 Conference*, 23-29, 1994.

26.Good, L., Bederson, B., Zoomable User Interfaces as a Medium for Slide Show Presentations, *Information Visualization*, 1 (1), Palgrave Macmillan, 35-49, 2002.

27.Greene, S., Tanin, E., Plaisant, C., Shneiderman, B., Mushlin, R., Olsen, L., Major, G., Johns, S, The End of Zero-Hit Queries: Query Previews for NASA's Global Change Master Directory, *International Journal of Digital Libraries*, Vol. 2, No. 2 (3), 79 - 90, 1999.

28.Huhns, M.N., Stephens, L.M., Agents on the Web: Personal Ontologies, *IEEE Internet Computing,* 3 (5), 85 - 87, 1999.

29.Isakowitz, T., Stohr, E., Balasubramanian, P., RMM: A Methodology for Structured Hypermedia Design, *Communications of the ACM*, 38(8), 34-44, August, 1995.

30.Jones, S., Graphical Query Specification and Dynamic Result Previews for a Digital Library. *Proceedings of ACM Symposium on User Interface Software and Technology (UIST98)*, 1998.

31.Kandogan, E., Shneiderman, B., Elastic Windows: Evaluation of Multi-Window Operations, *Proceedings of the ACM CHI '97 Conference*, 250 - 257, 1997.

32.Kang, H., Shneiderman, B., MediaFinder: An Interface for Dynamic Personal Media Management with Semantic Regions, *Extended Abstract of CHI2003,* ACM Press, 2003.

33.Kang, H., Shneiderman, B., Visualization Methods for Personal Photo Collections: Browsing and Searching in PhotoFinder, *Proceedings of International Conference on Multimedia & Expo,* 2000.

34.Krasner, G.E., Pope, S.T., A Cookbook for Using The Model-View-Controller User Interface Paradigm in Smalltalk-80, *Journal of Object-Oriented Programming*, 1(3), 26 - 49, 1988.

35.Kuchinsky, A., Pering, C., Creech, M. L., Freeze, D., Serra, B., and Gwizdka, J., FotoFile: A Consumer Multimedia Organization and Retrieval System, *Proceedings of the ACM CHI '99 Conference*, 496-503, May, 1999.

36.Kullberg, R.L., Dynamic Timelines: Visualizing Historical Information in Three Dimensions, *M.S. Thesis at the MIT Media Laboratory*, September, 1995.

37.Lamming, M., Forget-me-not: intimate computing in support of human memory, *FRIEND21, '94, International Symposium on Next Generation Human Interface*, Meguro Gajoen, Japan, 1994.

38.Long, B., TimeStore: Exploring Time-Based Filing. *Unpublished paper*, University of Toronto, 1994.

39.Mander, R., Salomon, G., Wong, Y.Y., A pile metaphor for supporting casual organization of information. *Proceedings of the ACM CHI92 Conference*, 627-634. 1992.

40.Mayo, R.N., Reprint of the Factoid Webpage, *Technical Note TN-60*, Compaq Computer Corporation, Western Research Laboratory, July, 2001.

41.Moore, G.E., Gigabits and Gigabucks, *University Video Corporation Distinguished Lecture,* 1996.

42.Nardi, B., Barreau, D., Finding and Reminding: File organization from the Desktop, *ACM SIGCHI Bulletin*, 27(3), 39-43, 1994.

43.Nardi, B.A., Whittaker, S., Isaacs , E., Creech, M., Johnson, J., Hainsworth, J., Integrating communication and information through ContactMap, *Communications of the ACM*, Volume 45, Number 4, pp. 89-95, 2002.

44.North, C., Shneiderman, B., Component-Based, User-Constructed, Multiple-View Visualization, *Extended Abstracts of CHI '2001*, 201-202, 2001.

45.Pepys Project at XRCE, http://www.xrce.xerox.com/programs/mds/past-projects/pepys.html

46.Placeless Documents Project Archive, http://www2.parc.com/csl/projects/placeless/

47.Plaisant, C., Carr, D., Shneiderman, B., Image browsers: taxonomy, guidelines, and informal specifications, *IEEE Software*, 12(2), 21-32, March, 1995.

48.Plaisant, C., Milash, B., Rose, A., Widoff, S., Shneiderman, B., LifeLines: Visualizing Personal Histories. *Proceedings of the ACM CHI96 Conference*, 221-227, 1996.

49.Plaisant, C., Rose, A., Rubloff, G., Salter, R., Shneiderman, B. The design of history mechanisms and their use in collaborative educational simulations, *Proceedings of the Computer Support for Collaborative Learning, CSCL' 99,* 348 – 359, Palo Alto, CA, 1999.

50.Quan, D., Bakshi, K., Huynh, D., and Karger, D. R., User interfaces for supporting multiple categorization. *In Proceedings of Interact 2003,* 2003.

51.Rao, R., Card, S., The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+Context Visualization for Tabular Information, *Proceedings of the ACM CHI94 Conference*, 318-322, 1994.

52.Rekimoto, J., TimeScape: A Time Machine for the Desktop Environment, *Proceedings of the ACM CHI '99 Conference*, 1999.

53.Ricardo, B., Berthier, R., *Modern Information Retrieval*, Addison Wesley, 1999.

54.Robertson, G., Czerwinski, M., Larson, K., Robbins, D. C., Thiel, D., van Dantzich, M., Data Mountain: Using Spatial Memory for Document Management, *Proceedings of ACM UIST98*, 153-162, November, 1998.

55.Rose, D.E., Borenstein, J.J., Tiene, K., MessageWorld: A new Approach to Facilitating Asynchronous Group Communication, *Proceedings in CIKM'95 (Conference on Information & Knowledge Management)*, 1995.

56.Roth, S., Lucas, P., Senn, J., Gomberg, C., Burks, M., Stroffolino, P., Kolojejchick, J., Dunmire, C., Visage: A User Interface Environment for Exploring Information, *Proceedings of Information Visualization*, IEEE, 3 - 12, 1996.

57.Roth, S.F., Kolojejchick, J., Mattis, J., Chuah, M., SageTools: An Intelligent Environment for Sketching, Browsing, and Customizing Data Graphics, *Proceedings of the ACM CHI95 Conference*, 409 - 410, 1995.

58.Shipman, F., Airhart, R., Hsieh, H., Maloor, P., Moore, J. M., and Shah, D., Visual and spatial communication and task organization using the visual knowledge builder. *In Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting GroupWork*, 260-269, 2001.

59.Shneiderman, B., Designing the User Interface: Strategies for Effective Human-Computer Interaction*,* Third Edition, *Addison-Wesley*, 1998.

60.Shneiderman, B., Inventing Discovery Tools: Combining Information Visualization with Data Mining, *Proceedings of Discovery Science 2001*, Springer-Verlag, Washington DC, 2001.

61.Shneiderman, B., Kang, H., Direct Annotation: A Drag-and-Drop Strategy for Labeling Photos, *Proceedings of International Conference Information Visualisation* (IV2000), 2000.

62. Shneiderman, B., Kang, H., Kules, B., Plaisant, C., Rose, A., Rucheir, R., A Photo History of SIGCHI: Design Evolution from Personal to Public, *ACM Interactions*, volume IX.3 May + June 2002, 17-23, ACM press, 2002.

63. Shneiderman, B., The Eyes Have It: A Task by Data Type Taxonomy of Information Visualizations, *Proceedings of IEEE Visual Languages '96*, 336-343, 1996.

64. Siviter D., Petre, M., Klein, B., Harnessing technology for effective inter- and intra-institutional collaboration, *Report of the ITiCSE'97 Working Group on Supporting Inter- and Institutional Collaboration*, 1997.

65. SpotFire, http://www.spotfire.com

66. Tang, L., Dynamic Aggregation, CMSC838 class project in spring 2001, http://www.cs.umd.edu/class/spring2001/cmsc838b/Project/Lida_Tang/index.html

67. The Diary Agent, *Project Web Page*, http://www.d.kth.se/~d93-jlu/Mobile/diaryagent.html

68. Thevenin, D., Coutaz, J., Plasticity of User Interfaces: Framework and Research Agenda, *Proceedings of INTERACT'99*, 1 – 8, 1999.

69. TinderBox, http://www.eastgate.com/Tinderbox/

70. Tufte, E., The Visual Display of Quantitative Information, *Graphics Press*, Cheshire CT, 1983.

71. Tufte, E., Visual Explanations, *Graphics Press*, 1997.

72. Uchihashi, S., Foote, J., Girgensohn, A., Boreczky, J., Video Manga: generating semantically meaningful video summaries, *Proceedings of the seventh ACM international conference on Multimedia,* 1999.

73. Video Diary Project at XRCE, http://www.xrce.xerox.com/programs/mds/past-projects/video-diary.html

74. Weiland. W.J., Schneiderman, B., A graphical query interface based on aggregation/generalization hierarchies, *Information Systems***,** Volume 18 , Issue 4, 215 – 232, 1993.

75. Williamson, C., Shneiderman, B., The Dynamic Home Finder: Evaluating Dynamic Queries in a Real-Estate Information Exploration System, *Proceedings of ACM SIGIR '92 Conference*, 338 - 346, 1992.

76.Wolff, G. J., Barrus, J., Multimedia Conversations with Point & Talk Interfaces, *TR*, Ricoh California Research Center, 2001.

77.Woodruff, A., Faulring, A., Rosenholtz, R., Morrison, J., and Pirolli, P., Using Thumbnails to Search the Web, *Proceedings of ACM CHI2001,* 198–205, April, 2001.

78.Zellweger, P., Mackinlay, J., Good, L., Stefik, M., and Baudisch, P., City Lights: Contextual Views in Minimal Space. *Extended Abstract of CHI2003 ACM Conference on Human Factors in Computing Systems,* 2001.

79.Rao, R., Card, S.K., The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+Context Visualization for Tabular Information, *Proceedings of the ACM CHI94 Conference on Human Factors in Computing Systems*, 318-322, April, 1994.

Appendix A:

Data Structure

A.1 SemanticRegion Class Structure

```
Public Class SemanticRegion
    Inherits System.Windows.Forms.UserControl

    Public Const Border = 20                'Constant - Size of Border

    Public ParentSR As SemanticRegion        ' pointer to parent region

    Public DragSize As Boolean = False       'Resize - Drag Handle of Region
    Public StartDragPt As Point              'Resize Semantic Regions - Start Point
    Public EndDragPt As Point                'Resize Semantic Regions - End Point
    Public StartDrag As Boolean = False

    ' Zooming Members
    Public ZoomIndex As Integer
    Public ZoomCount As Integer = 0
    Public numTrans As Integer = 15
    Public StartRect, FinalRect As Rectangle
    Public Maximized As Boolean = False
    Public PrevLocation As Point = New Point()
    Public PrevSize As Size = New Size()

    'Recursion
    Public ChildSemanticRegionList As ArrayList = New ArrayList()
    Public PersonalMediaItemList As ArrayList = New ArrayList
    Public SelectedChildSemanticRegionList As ArrayList = New ArrayList()
    Public MainSelectedRegion As SemanticRegion
    Public RemainingItem As SemanticRegion

    'Select and Move Child Semantic Regions
    Public DragChildSR As Boolean = False
    Public DragChildSRStart As Point
    Public DragChildSREnd As Point

    'Scale Semantic Regions
    Public RelativeLocation As Point = New Point()
    Public RelativeSize As Size = New Size()
    Public ManualResize As Boolean = False
    Public ScaleRect As Rectangle

    'Semantic Data of the Region
    Public SemanticData As Object

    'Arrange Items
    Public ItemCol As Integer = 1
    Public ItemRow As Integer = 1

    'Dynamic Intersection
    Public IntersectionArray(,) As Boolean
    Public IntersectionCountArray(,) As Integer

    'Boolean operation
    Public Level As Integer = 0
    Public BooleanOP As Integer = -1
    Public BooleanOperands As ArrayList = New ArrayList()

    'Data Items
    Public SelectedItems As ArrayList = New ArrayList()

    'Other Features
    Public SRShape As Integer = Shape.Rectangle
    Public LineList As ArrayList = New ArrayList

    Public BackImageFile As String = ""
    Public BackImage As Image
    Public SRView As Integer = Globals.PanelView.ItemView
End Class
```

A.2 FileAttributeInfo Class used for specifying the semantics of the region

```
Public Class FileAttributeInfo
    Public Name As String
    Public Path As String
    Public Extension As String
    Public CaseSensitive As Boolean = False
    Public DontCareSize As Boolean = True
    Public SizeFrom As Long
    Public SizeTo As Long
    Public DontCareCreated As Boolean = True
    Public CreatedOption As Integer
    Public CreatedMonth As Integer
    Public CreatedDay As Integer
    Public CreatedDateFrom As Date = Date.Now
    Public CreatedDateTo As Date = Date.Now
    Public DontCareModified As Boolean = True
    Public ModifiedOption As Integer
    Public ModifiedMonth As Integer
    Public ModifiedDay As Integer
    Public ModifiedDateFrom As Date = Date.Now
```

```
        Public ModifiedDateTo As Date = Date.Now
        Public DontCareAccessed As Boolean = True
        Public AccessedOption As Integer
        Public AccessedMonth As Integer
        Public AccessedDay As Integer
        Public AccessedDateFrom As Date = Date.Now
        Public AccessedDateTo As Date = Date.Now
        Public Archive As CheckState = CheckState.Indeterminate
        Public Hidden As CheckState = CheckState.Indeterminate
        Public RO As CheckState = CheckState.Indeterminate
        Public System As CheckState = CheckState.Indeterminate
        Public Compressed As CheckState = CheckState.Indeterminate
        Public Temporary As CheckState = CheckState.Indeterminate
        Public Title As String
        Public Subject As String
        Public Comment As String
        Public Custom As ArrayList
        Public Category As ArrayList = New ArrayList
        Public People As ArrayList = New ArrayList
    End Class
```

A.3 PersonalMediaItem Class representing a personal media item that has the form of a file

```
Public Class PersonalMediaItem
    Public InfoChanged As Boolean = False
    Public Title As String
    Public Subject As String
    Public Comments As String

    Public People As ArrayList = New ArrayList
    Public Category As ArrayList = New ArrayList
    Public LVItem As ListViewItem = New ListViewItem

    Public Thumb As Image
    Public Path As String
    Public FI As FileInfo
End Class
```

A.4 Semantic Regions Template file structure

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<Main>
  <BackgroundImage>C:\users\kang\ usa-map.jpg</BackgroundImage>
```

```
<SRList>
  <Number>5</Number>
  <SemanticRegion>
    <SRInfo>
      <Name>California</Name>
      <X>9</X>
      <Y>394</Y>
      <Width>141</Width>
      <Height>320</Height>
      <BackgroundImage>C:\Users\Kang\MediaFinder\bin\ca.jpg</BackgroundImage>
      <Shape>0</Shape>
      <View>8</View>
      <Color>-4128832</Color>
      <TextAlign>MiddleLeft</TextAlign>
      <FontName>Microsoft Sans Serif</FontName>
      <FontSize>8.25</FontSize>
      <FontBold>False</FontBold>
      <FontItalic>False</FontItalic>
      <Level>0</Level>
      <BooleanOp>-1</BooleanOp>
      <numBooleanOperands>0</numBooleanOperands>
    </SRInfo>
    <SRData>
      <Name>NULL</Name>
      <Path>NULL</Path>
      <Extension>NULL</Extension>
      <CaseSensitive>False</CaseSensitive>
      <DontCareSize>True</DontCareSize>
      <SizeFrom>0</SizeFrom>
      <SizeTo>0</SizeTo>
      <DontCareCreated>True</DontCareCreated>
      <CreatedOption>3</CreatedOption>
      <CreatedMonth>0</CreatedMonth>
      <CreatedDay>0</CreatedDay>
      <CreatedDateFrom>5/25/2003</CreatedDateFrom>
      <CreatedDateTo>5/25/2003</CreatedDateTo>
      <DontCareModified>True</DontCareModified>
      <ModifiedOption>3</ModifiedOption>
      <ModifiedMonth>0</ModifiedMonth>
      <ModifiedDay>0</ModifiedDay>
      <ModifiedDateFrom>5/25/2003</ModifiedDateFrom>
      <ModifiedDateTo>5/25/2003</ModifiedDateTo>
      <DontCareAccessed>True</DontCareAccessed>
      <AccessedOption>3</AccessedOption>
      <AccessedMonth>0</AccessedMonth>
      <AccessedDay>0</AccessedDay>
      <AccessedDateFrom>5/25/2003</AccessedDateFrom>
      <AccessedDateTo>5/25/2003</AccessedDateTo>
      <Archive>2</Archive>
      <Hidden>2</Hidden>
      <RO>2</RO>
      <System>2</System>
      <Compressed>2</Compressed>
      <Temporary>2</Temporary>
      <Title>NULL</Title>
  <Subject>NULL</Subject>
  <Comment>NULL</Comment>
  <People>
    <numPeople>0</numPeople>
  </People>
  <Category>
    <numCategory>1</numCategory>
    <Cat>Location</Cat>
    <Val>USA\CA</Val>
  </Category>
</SRData>
<ChildSemanticRegions>
  <numChild>1</numChild>
  <SemanticRegion>
    <SRInfo>
      <Name>Los Angeles</Name>
      <X>312</X>
      <Y>509</Y>
      <Width>124</Width>
      <Height>175</Height>
      <BackgroundImage>NULL</BackgroundImage>
      <Shape>0</Shape>
      <View>1</View>
      <Color>-4128832</Color>
      <TextAlign>MiddleLeft</TextAlign>
      <FontName>Microsoft Sans Serif</FontName>
      <FontSize>8.25</FontSize>
      <FontBold>False</FontBold>
      <FontItalic>False</FontItalic>
      <Level>0</Level>
      <BooleanOp>-1</BooleanOp>
      <numBooleanOperands>0</numBooleanOperands>
    </SRInfo>
    <SRData>
      <Name>NULL</Name>
      <Path>NULL</Path>
      <Extension>NULL</Extension>
      <CaseSensitive>False</CaseSensitive>
      <DontCareSize>True</DontCareSize>
      <SizeFrom>0</SizeFrom>
      <SizeTo>0</SizeTo>
      <DontCareCreated>True</DontCareCreated>
      <CreatedOption>3</CreatedOption>
      <CreatedMonth>0</CreatedMonth>
      <CreatedDay>0</CreatedDay>
      <CreatedDateFrom>5/25/2003</CreatedDateFrom>
      <CreatedDateTo>5/25/2003</CreatedDateTo>
      <DontCareModified>True</DontCareModified>
      <ModifiedOption>3</ModifiedOption>
      <ModifiedMonth>0</ModifiedMonth>
      <ModifiedDay>0</ModifiedDay>
      <ModifiedDateFrom>5/25/2003</ModifiedDateFrom>
      <ModifiedDateTo>5/25/2003</ModifiedDateTo>
      <DontCareAccessed>True</DontCareAccessed>
      <AccessedOption>3</AccessedOption>
```

```
            <AccessedMonth>0</AccessedMonth>
            <AccessedDay>0</AccessedDay>
            <AccessedDateFrom>6/4/2003</AccessedDateFrom>
            <AccessedDateTo>6/4/2003</AccessedDateTo>
            <Archive>2</Archive>
            <Hidden>2</Hidden>
            <RO>2</RO>
            <System>2</System>
            <Compressed>2</Compressed>
            <Temporary>2</Temporary>
            <Title>NULL</Title>
            <Subject>NULL</Subject>
            <Comment>NULL</Comment>
            <People>
              <numPeople>0</numPeople>
            </People>
            <Category>
              <numCategory>1</numCategory>
              <Cat>Location</Cat>
              <Val>USA\CA\Los Angeles</Val>
            </Category>
          </SRData>
          <numChild>0</numChild>
          <Lines>
            <Number>0</Number>
          </Lines>
        </SemanticRegion>
      </ChildSemanticRegions>
      <Lines>
        <Number>0</Number>
      </Lines>
    </SemanticRegion>
  <SRList>
  <Lines>
    <Number>0</Number>
  </Lines>
</Main>
```

Appendix B:

Usability Study Survey

B.1 Background Survey

1. Occupation (Major, Grad vs. Undergrad, Year)
2. Gender and Age
3. Computer Usage Experience and main usage

   (How long have you been using computers?, How many different computer packages have you probably used ?)
4. Window File System Concepts
5. Object-Oriented Drawing Tool Experience
6. Visualization Tools (Data Analysis Tools) Experience
7. Programming Experience
8. Internet Access (what kind?, how long?)
9. Personal Media Data (type of personal media you have, e.g. Video, Audio, Photo, Ebook)
10. Personal Digital Photo Experience
11. How to manage photos (name of a tool)

B.2 Verbal Post Survey for First User Study

1. Other ideas for exploring personal photo data (or personal media data such as email, files, and so on)
2. Comments on the *fling-and-flock* metaphor
3. Trouble spots in using MediaFinder
4. Suggestions for improving MediaFinder user interface
5. Ease of use:

Difficult 1 2 3 4 5 6 7 8 9 Easy

6. Willingness to use MediaFinder for your photos:

(With given metadata)

Not Interested 1 2 3 4 5 6 7 8 9 Willing to use

(With metadata annotated by yourself)

Not Interested 1 2 3 4 5 6 7 8 9 Willing to use

B.3 Verbal Post Survey for Second User Study

1. General comments on the construction of *Semantic Regions*

7. Trouble spots in using MediaFinder

8. Suggestions for improving MediaFinder user interface

9. Ease of use:

Difficult Easy

1 2 3 4 5 6 7 8 9

10. Willingness to use MediaFinder for your photos:

(With given metadata)

Not Interested Willing to use

1 2 3 4 5 6 7 8 9

(With metadata annotated by yourself)

Not Interested Willing to use

1 2 3 4 5 6 7 8 9

11. How helpful was the MediaFinder interface ?

Disturbs Helps

1 2 3 4 5 6 7 8 9

12. Would you prefer MediaFinder to the tools you already use ?

Not at all Certainly

1 2 3 4 5 6 7 8 9

B.4 List of Measuring Criteria for the Second User Study

- Learning time (for construction)

- Success (y/n how close to success) and Time to completion

1 (Time:)

2 (Time:)

3 (Time:)

4 (Time:)

- Cognitive trouble spots (in training and test trials)

- MediaFinder interface problems.