

---

# Unified Associative Information Storage and Retrieval

**Brandon Rothrock**

Human-Computer Interaction  
Institute  
Carnegie Mellon University  
rothrock@cs.cmu.edu

**Brad A. Myers**

Human-Computer Interaction  
Institute  
Carnegie Mellon University  
bam@cs.cmu.edu

**Sophie H. Wang**

Center for Automated Learning  
and Discovery  
Carnegie Mellon University  
sophie.wang@cs.cmu.edu

**Abstract**

We present a novel system for performing information management in a unified manner. Users currently must manage large amounts of data which may be fragmented across file formats and applications. Our system, called Iolite, attempts to consolidate this information by automatically discovering associations within the data. Iolite uses these associations to provide a unified interface to navigate and operate on this information space.

**Keywords**

Information management

**ACM Classification Keywords**

I.3.6. Interaction techniques

**Introduction**

Knowledge workers are faced with managing increasing amounts of digital information. This information may be fragmented across file formats, storage location, and the tools used to manage it. For example, email attachments can be stored both within the email application and the file system. Similarly, web links can be stored within email, the web browser, and the file system. Managing these fragments as they move and change between tools is often done manually. The

---

Copyright is held by the author/owner(s).  
CHI 2006, April 22-27, 2006, Montréal, Québec, Canada.  
ACM 1-59593-298-4/06/0004.

widespread use of the email client to manage tasks and appointments in addition to email further complicates this task by making it difficult to identify one type of information from another [8].

### **Information Management**

The two primary tasks of any information management system are facilitating *organization* and *recall*.

Organization is the structuring of information under a scheme that can be consistently reproduced by the user. Recall is the ability to locate a specific piece of information. The following approaches represent several systems designed to improve either organization or recall.

The most prevalent form of information management is manual organization. There are large variations of organization strategies in common use. For example, email users are often classified into frequent-filers, spring-cleaners, and no-filers [8, 2]. Similarly, users of the file system organize by hierarchy, time, or spatial position on the desktop [6, 7]. Users that choose to methodically organize have the benefit of fast manual recall. This organization process is time consuming, however, and many users choose not to organize for this reason. These no-filers must either employ the use of tools to recall their data or spend considerable time manually locating this information.

Many tools are available to assist with automatic organization of data. Lifestreams [4] automatically organizes all information by time. This alleviates the burden of manual filing and has the advantage of a consistent and unambiguous organization for many different types of information. Some of the difficulties with Lifestreams include recalling information with no

clear timestamp, such as contacts and addresses. It is also difficult to recall densely packed information using this system. For example, it is not uncommon for users to receive several hundred emails a day. Specifying a time range specific enough to recall a single item may not be possible in this case. This may limit the effectiveness of this technique as information volume grows.

Content-based search has gained widespread appeal as a universal tool for information access. Tools such as Google Desktop Search and Apple Spotlight have shown considerable success as personal information management tools. These tools operate without any restriction on the user's organization of data. With an appropriate search query, these tools can often recall information quickly and accurately. When the desired information cannot be immediately found, the user must refine the search query or manually search through a list of results. One of the drawbacks of this technique is that the user receives very little feedback on the quality of their search query. When items are not found, the user may waste considerable time trying to refine the query.

A common problem with all of these techniques is the inability to handle data fragmentation. Suppose a user had authored a document using information from many sources such as email, spreadsheets, and web pages. Each piece of information may have been processed at different times, be of different types, and stored in different locations. None of the above techniques provide any direct way of identifying this information.

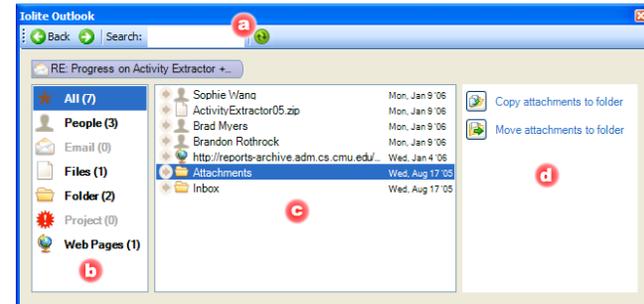
## An Associative Approach

We present our design for an information management system called Iolite or Intelligent Qn-Line Interferencing for Text and Email. Iolite attempts to use information association to provide a unified user interface for both organization and recall of all types of information.

Knowledge workers spend a considerable amount of time producing and consuming information. Such tasks may include composing a document from an email request, or creating a distribution list from a collection of contacts. The relationship between this information is rarely tracked by information management tools, yet this context may be highly relevant in locating this information at a future time.

Iolite utilizes a set of agents to discover these associations by observing the user and monitoring activities on the computer. Many associations can be observed directly, such as a saved attachment from an email message and files that share data from the clipboard. Other associations can be inferred through machine learning or other intelligent techniques, such as extracting projects and tasks from email messages.

Studies have shown that users are most comfortable finding information by *orienting* [1], or in other words steering toward the desired information in a series of iterative steps instead of jumping directly there. This is because the user likely possesses contextual knowledge about what they're looking for that is not easily expressed in a query. This interaction is used in Iolite by allowing the user to freely navigate the network of associations.



**figure 1:** Iolite running within Microsoft Outlook. Context sensitive tools are shown on the right.

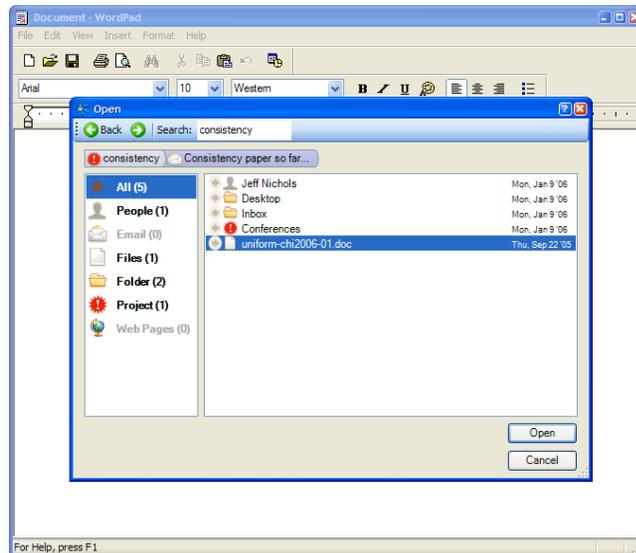
## User Interface

The Iolite user interface incorporates a browser style design. The Iolite client for Microsoft Outlook (figure 1) will automatically display all information associated with the email that is currently selected in Outlook. By clicking the arrow next to any entity in the center pane (figure 1.c), Iolite will navigate to that entity and display its associations. The left pane (figure 1.b) contains a list of filters. Selecting a filter will display only associations of that type. Iolite for Outlook also provides context sensitive tools shown on the right (figure 1.d) to assist with common tasks.

The view in figure 1 shows the associations of an email containing an attachment. Iolite has identified that email within the inbox often have their attachments stored into the "Attachments" folder and has associated the email to both the inbox and the file system folder "Attachments". One of the context-sensitive rules has placed two tools in the right pane (figure 1.d) to copy and move the attachments to the selected folder.

Iolite can also be used to replace the standard Windows dialogs for opening and saving files (figure 2). This allows Iolite to be the default file manager for most applications without modification. Additionally, Iolite can be used as a standalone client to locate and open files and resources similar to Windows Explorer.

Context can be shared between all applications by toggling the synchronize button (figure 1.a) in Iolite. Upon synchronization, all Iolite clients will display the same contents and begin tracking each other if any navigation occurs. This can be used to maintain a consistent working path when managing files or resources between separate applications.



**figure 2:** Iolite replaces the Windows file open dialog within WordPad.

### Usage Example

In this section we describe how Iolite can be used through a sample task using both Iolite and traditional tools. The task involves two different emails from the same sender: one email contains a WordPad document and the other contains an image that must be added to that document. The modified document must then be attached to an email and sent to all individuals involved in the Iolite project. The user begins the task by viewing the email containing the WordPad document in Outlook.

The Iolite for Outlook window will then display associations of the selected email which will include the folders for the project associated with the email. The user starts by saving the attachment, which is done by selecting an associated folder and clicking the “Copy attachment to folder” tool. In addition to saving the file, a new association will appear in the Iolite window representing the local copy of the attached document. The user then double-clicks on this new association to launch WordPad and opens the document.

Now the user wants to insert an image into the document. In WordPad, the user opens the insert object dialog and selects the browse button to find an image. This opens the embedded Iolite Open dialog (see figure 2). All Iolite clients are synchronized to each other, so the dialog will already be navigated to the document email that we last accessed through the Iolite for Outlook window. The user can click on the arrow next to the sender’s name, which will display a list of entities related to the sender. If the list is too large, the user can select the email filter to view only emails related to the sender. Once the second email is found, the attachment appears and can be opened.

Iolite will automatically extract the file from the email, save it to a temporary location if necessary, and provide the correct path to WordPad.

Adding the image completes the document, which must now be sent to all of the people on the Iolite project. To send the email, the user switches to the Iolite for Outlook window, and selects the associated project that Iolite has inferred for this email. A tool will appear in the right pane to send an email to everyone on this project. Clicking the tool displays a new email window with the "To:" field already filled in with the appropriate names. The user can then complete the task by attaching the document and clicking "Send."

Using traditional tools, this process would be more complicated. To start, the user would have to manually navigate through the file system to find the appropriate folder in which to save their document. After saving the document, they will then have to repeat this navigation to find the document so it can be opened in WordPad. The user must then find the email containing the image to add to the document. Not only could the process of finding this email be long and tedious, but the user will then again be forced to navigate through their file system two more times: once to save the attachment to an appropriate place and again to insert the object into their document. Many users will avoid this repetitive navigation by saving all of their files in one location that can be navigated to easily, such as the Desktop on Microsoft Windows. Using this strategy does decrease the time of our example task, but it also results in a mess of temporary files that must eventually be organized or deleted. We believe that the cost of navigating in Iolite is low enough that users

would no longer need to use messy shortcuts of this form.

Another complication for traditional tools is finding the complete set of people for a particular email. In email clients, users would need to manually search their message folders and record one-by-one the set of people to whom the message should be sent. Iolite's associations provide an easy way to automatically aggregate recipients for an email message without the need for manual searching or the maintenance of multiple distribution lists.

### **The Association Database**

Associations are represented as an undirected graph stored in a relational database. This database is populated by a collection of agents that continuously observe the user's activity. In the event that an association is not observable, Iolite also provides an interface for the user to manually insert, modify, or delete an entity or association. Our current prototype contains agents for Microsoft Outlook and the operating system.

The Outlook agent primarily records direct associations found in email. These associations include mapping an email with its sender, recipients, extracted URLs, attachments, etc. This agent also contains the ActivityExtractor classifier [5]. ActivityExtractor uses a Naïve Bayes and EM algorithm to cluster email messages based on their word distributions. A social network analysis of email senders and recipients is then performed to infer what project the email belongs to. This project is then associated with the classified email.

The operating system agent monitors the file system and associates files with their folders. The agent also monitors the clipboard and associates files that have had contents copied and pasted between them.

### Future Work

We plan to perform a field study by publicly deploying an instrumented version of Iolite and evaluating performance under real use. We believe this will be more informative than a lab study due to the large variation of individual use.

We also plan on further exploring the use of machine learning to classify new associations such as task flows similar to the ones used in TaskTracer [3]. Additionally we are investigating how user feedback can be used to better train the classifiers and provide corrections for misclassifications.

### Conclusions

Information management is becoming increasingly more difficult as the volume of information users need to process grows. Iolite attempts to address this problem by intelligently discovering associations within the data and presenting the user with a single, unified interface to navigate this information. We believe the simplicity and extensibility of this tool provides for an efficient interface for large amounts of information, and could potentially perform well against manual organization and content-based search.

### Acknowledgements

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. NBCHD030010.

### References

- [1] Alvarado, Christine; Teevan, Jaime; Ackerman, Mark S.; and Karger, David. Surviving the Information Explosion: How People Find Their Electronic Information. MIT Artificial Laboratory Technical Report, April 2003.
- [2] Bälter, O. (1997). Strategies for organising email messages. In HCI. Springer, London, UK, 21–38.
- [3] Dragunov, A. N., Dietterich, T. G., Johnsruide, K., McLaughlin, M., Li, L., and Herlocker, J. L. (2005). TaskTracer: A desktop environment to support multi-tasking knowledge workers. Proceedings of IUI.
- [4] Freeman, E and Gelernter, D (1996). Lifestreams: A storage model for personal data. ACM SIGMOD Bulletin 25, 80-86.
- [5] Mitchell, T., Wang, S., Huang, Y., Cheyer, A. (2005). Inferring Users' Projects from their Workstation Contents. Carnegie Mellon University Technical Report CMU-CALD-05-113.
- [6] Nardi, B. and Barreau, D. (1997). "Finding and reminding" revisited: appropriate metaphors for file organization at the desktop. ACM SIGCHI Bulletin 29, 1, 76-78.
- [7] Nardi, B., Whittaker, S. and Bradner, E. (2000). Interaction and outeraction: instant messaging in action. In Proceedings of CSCW '00, 79-88.
- [8] Whittaker, S., Sidner, C. (1996). Email overload: Exploring personal information management of email. ACM SIGCHI.