net SCALE Technologies, Inc.

281 State Hwy 79, Morganville, NJ 07751, Tel. 732-970-1441, http://www.net-scale.com

A Solution for Personal Document Storage

White Paper

Version: 1.0
Last modified: August 20, 2005
Author: Urs A. Muller
Distribution: “A” (Approved for Public Release, Distribution Unlimited)

Summary

While in recent years bandwidth, connectivity and world wide data communication have become cheap and ubiquitous, the problem of storing and preserving personal digital data is still largely unsolved. Most users struggle with preserving data, moving data from one PC generation to the next, finding personal documents on one’s own hard drive and sharing personal documents with others. This paper introduces a solution based on network storage which also integrates e-mail, document indirection similar to DNS and a unified directory tree for e-mail and files.

1. Introduction

"The nation's 115 million home computers are brimming over with personal treasures – millions of photographs, music of every genre, college papers, the great American novel and, of course, mountains of e-mail messages.

Yet no one has figured out how to preserve these electronic materials for the next decade, much less for the ages. Like junk e-mail, the problem of digital archiving, which seems straightforward, confounds even the experts."

The above two paragraphs are quoted from a recent New York Times article [1]. The article describes everyday problems consumers face around personal data management. While technical solutions to any one particular problem do exist, end users don't normally benefit from those because the solutions are too complex and labor intensive to be carried out by end users on a regular basis. We have categorized the range of problems as follows:

- *Migration:* How to copy personal documents and e-mail from an old computer to a new one.
- *Backup:* How to make sure no important personal data is lost when a computer hard disk crashes or a laptop is lost or stolen.
- *Access:* How to access one's personal documents and e-mail when away from the personal computer.
- *Search:* How to find information which may date back several decades.
- *Sharing:* How to let other people access selected personal documents and e-mail messages in a controlled and secure way.
- *Preservation:* How to make sure the data survives the aging of disks, tapes, CDs, and DVDs and remains accessible even though disk technology and file systems are changing.

Furthermore, for most people, e-mail and documents are stored in completely separate directory trees with different and mutually incompatible access methods. The data may reside on the same disk but the software needed to access and manage it is completely different with distinct user interfaces (UIs). For example many users will use the Windows Explorer to browse and manage documents while using Outlook to browse and manage e-mail. This separation between e-mail and other files seems to be arbitrary and useless.

As mentioned earlier, these are not unsolved problems. Technical solutions to any one of them exist. However, making these solutions work in everyday life requires knowledge, skill, and time, a combination of resources which most people don't have or are not willing to afford.

Just because solutions to a particular problem exist doesn't automatically benefit the general public. For example, when building a new house in a remote location, technology exists to bring electrical power to the house right away, for example, through the purchase and operation of a generator. However, most people would probably not want to get involved in selecting, installing, and maintaining a generator and would rather wait for the electrical company to connect the house to the public power grid. Having power generation managed by companies serving thousands or millions of customers has significant advantages despite the additional cost of building and operating a power grid and the power lines to each home.

It seems natural that similar advantages can be gained by providing managed solutions for personal document storage. This paper presents a specific solution for this. The next section documents the current problems in more detail and the section following describes the proposed solution. The paper ends with some conclusions.

2. The Current Reality

This section analyzes the list of problems introduced in the previous section in more details.

2.1. Migration

In this context, migration means moving data from an old computer to a new one and converting old document formats into new ones. In other words, taking all the necessary steps to ensure data remains accessible and readable by a user's current computer system. Some illustrative real life stories about this topic can be found in [1]. When users switch, for example, from an old word processor to a new one or from an old computer system to a new one, they typically leave the old data in its current format and at its current location. As a consequence, users are afraid to give away, sell, or dispose of old computer equipment because this would mean that they loose old data. Many users are still with their first or second generation computer so this problem is not very severe yet. However, if extrapolating over a person's life time the situation looks grim. Computers, monitors, and keyboards will pile up in closets.

2.2. Backup

Good data backup requires storing a copy of all data regularly and frequently at a location other than where the main data storage resides. Furthermore, backed up data needs to be temporarily restored in dry runs on a regular basis and checked for consistency. This is to ensure the backup process actually works and that no unnoticed bugs sneak into the backup procedures or software over time. This is

a scenario which is practiced by only very few users; most users just give up and tolerate the possibility of a complete data loss.

2.3. Access

In the past, stored data could only be accessed from the computer system to which the storage was attached. Despite the fact that the Internet is now prevalent and broadband data connections are available in many households, this is, for the most part, still true. There is no lack of software which makes PCs accessible from remote locations but it is left up to the end user to install and securely configure, and to maintain their PCs as servers, e. g., to configure them such that they automatically restart with all the necessary software after a power outage. Peer-to-peer software was somewhat successful in changing this. However, while it has achieved other goals, it has not helped end users accessing their own personal documents when away from their PCs. Finally, a large number of file sharing servers exist today, mostly intended to share photos. Such services seem to make it easier for users to share data with others but they seem to do little to support general remote access to personal data.

2.4. Search

Surprisingly, today it is easier to find a particular piece of information within the petabytes of data on the Internet than it is to find information which dates a few years back on one's own PC. This is because the large Internet search engines, Google probably being the best known, have been tremendously successful in indexing the entire Internet and creating software to do so while PC users mostly use fairly simple and inefficient search tools for their own data. In principle, the Internet search technology is available also to PC users. For example, Google just recently released their Desktop Search software which seems to put Google-type search capabilities on a user's PC. However, many users will likely not choose to install, configure and maintain this software. In our own trial, for example, it took three days to create the initial index on a PC.

2.5. Sharing

Sharing personal documents is another deceptively easy task. The greatest need in this area is probably for digital photographs. They can be burnt on CDs or DVDs and mailed, they can be e-mailed, or posted on a picture share service. Unfortunately, end users seem to struggle with any of these methods. Burning CDs or DVDs is inconvenient, the disks may get damaged during transport and they may be written in a format which is unreadable by the recipient's PC. Sending pictures by e-mail usually is limited by mailbox sizes and can cause angry reactions by

recipient's whose mailboxes have just been clogged up by pictures. Picture share services require time to setup and manage. Furthermore, it is either complicated or impossible to manage access control.

2.6. Preservation

Today's technology allows for storage and quick access of vast amounts of data but preserving the data is still a problem. Most forms of digital data storage will not hold the data for more than a few decades, and, to our knowledge, no technology exists which can preserve data over several centuries. The only way to preserve data over long periods of time is by regularly copying it from one medium to another, in other words, "refreshing" the data.

While this task is manageable by corporations and government entities, such as large libraries, it is a nearly impossible task for individual users. While, for example, paper photographs can be put in a shoe box and, after 70 years, have still a very good chance of being viewable in acceptable quality, digital pictures placed on a CD ROM or hard disk and stored in a closet, will most likely not be accessible after 70 years, unless the user made a fresh copy of the data approximately every three to five years.

3. A Solution

3.1. Paradigm Shift to Network Storage

As outlined in the previous section, dealing with the wealth of problems surrounding personal document storage is very time consuming and for most end users it is just too overwhelming.

If the current paradigm of using the personal computer as the primary storage of electronic documents would be shifted to using network storage as the primary location, then a small number of professionals could handle all problems discussed in the previous section for the end user. We will refer to the proposed network storage solution as the *Personal Document Storage* or *PDS* from now on.

The PDS would not only be a great relief for end users, it also makes economic sense as a few professionals can do the job for millions of users. This is much more efficient than having millions of users doing the same job for themselves individually. This paradigm shift is illustrated in Figure 1. The PC hard drive changes its function from being the primary storage to being just a temporary workspace or it may no longer be used at all, depending on the quality of the Internet connection and user preferences.

Of course, this is quite a fundamental paradigm shift and it will take time for the professional world to provide the necessary tools and services and for end users to make a change in their habits. However, this would not be the first time such a

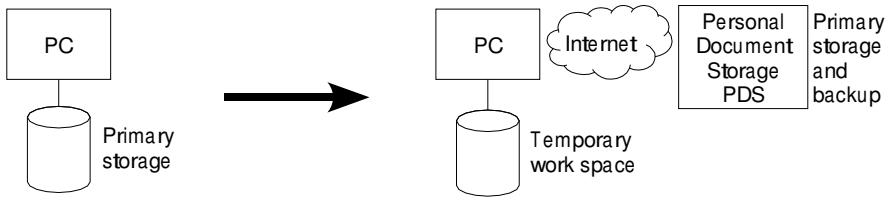

Figure 1: The paradigm shift from the personal computer being the primary storage location of electronic documents to network storage (Personal Document Storage or PDS) being the primary location.

paradigm shift occurs. One example is money. A few hundred years ago money was mostly kept in form of coins at home. Each person had to deal with the problems of storage, security, transport, and access on their own. Today most of the money that is earned and spent is never physically touched by the end users. It goes from employer electronically to the employee's bank account from there to vendors and service providers via use of credit card or electronic bill payment. We take it for granted, that we can check our account balance and access our money from almost anywhere in the world, and that we are able to transfer money to anybody we wish at any time and from any location. Credit cards, ATM machines, and on-line banking are among the technologies which make this possible. If we compare the ease with which we can access and transfer money today and how well the protection against misuse works with the current state of personal document storage it becomes apparent that personal document storage has a long way to catch up.

Until recently a solution wasn't necessary because most people had little or no electronic documents to store, share, and preserve. With the popularity of e-mail and the availability of cheap digital still and movie cameras this is rapidly changing, however. A successful network storage solution requires:

- Broadband Internet access being readily available to end users. The term "broadband" in this context means the availability of sufficient bandwidth to allow for the handling of an end user's daily data exchange needs without introducing much noticeable delay.
- An Internet with sufficient bandwidth capacity.
- Storage technology with sufficient storage size and access speed.
- A business model which supports service providers while at the same time preserving an end user's right to freely choose their provider without fear of loosing data or fundamental features.

The first three requirements are essentially satisfied today. A technical solution for the last requirement is introduced in Section 3.6.

3.2. E-Mail Integration

A substantial amount of personal data consists of e-mail. Traditionally, e-mail is received by a mail server operated by a service provider. Users download e-mail from the mail server to their PC and then back it up or archive it (or do no backup at all), as shown in Figure 2.

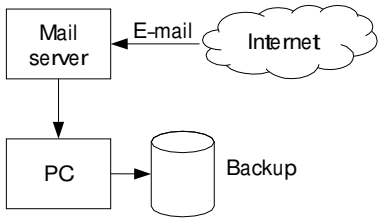

Figure 2: Traditional e-mail model. E-mail is received by a mail server, then down-loaded to the user's PC from where it may or may not be backed up.

It seems natural to extend the network storage with e-mail interfaces and thus allow it to serve the dual purpose of document storage and mail server. The resulting new model is shown in Figure 3. With this small change, e-mail gets archived automatically without a user's PC being on-line or even turned on and e-mail also benefits from all other network storage advantages which may be implemented, such as access, sharing, and search.

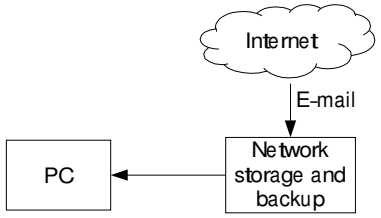

Figure 3: The repository extended with e-mail interfaces, so it serves the dual purpose of document storage and mail server.

3.3. Unified Directory Tree

It seems that most people have accepted the fact that e-mail resides in a totally different foldering system than all other documents. At least that is the case for most users and most of them probably would have difficulties imagining anything else. However, taking a closer look at the situation reveals that e-mail foldering systems are not a natural choice at all.

All popular operating systems today are equipped with a hierarchical file system whose features are so similar that single file systems can be shared among different

operating systems (Samba just being one example of a tool for this). The smallest unit of a file system is a file. Users and application programs decide what exactly goes into a file, for example, a single picture, multiple pages of a document, or a spread sheet. Files are organized in directories which in turn can be part of higher level directories, etc. The result is a directory tree. Users have almost complete freedom in organizing a directory tree.

A natural extension to the e-mail world would be to store each e-mail in a separate file. There is already an open data format for e-mail messages which is used by almost all e-mail systems which is defined in RFC 2822 [2]. That file format is already readable by many of the popular e-mail clients, such as Outlook Express and Firefox (not by Outlook, though). Nevertheless, most e-mail client creators have chosen a different route. E-mail messages are stored in internal proprietary folders instead of in directories of the file system. The proprietary folders are then packed into a single file or a few files in a proprietary format and stored deep down in the directory tree branch of the application settings. Most users probably don't know such files even exist much less have the knowledge nor patience to find them and back them up or move them on to a new PC. Since e-mail clients implement their own foldering system they also have to re-implement all foldering functionality, such as the entire UI for navigating through folders, import and export, copy, move, and search. While regular documents can be browsed, searched, and managed with a great variety of different tools from different vendors, e-mail can typically only be viewed and managed with exactly one proprietary application.

If, for example, digital pictures had gone the same route as e-mail did, then todays cameras would store all pictures in a proprietary format into one or a few files which could be viewed, managed, and modified with essentially only one dedicated application, e. g., from the camera manufacturer. Instead, today, pictures can be managed with any number of file browsers. Most of them are capable of displaying thumbnail views of the pictures which is a great help organizing pictures. In addition there is a wealth of image and slide show viewers available in addition to a multitude of picture editing, photo album, and print tools. Why couldn't there be, for example, a thumbnail view of e-mail messages in a regular file browser? Each icon could for instance contain the sender's name and subject information and maybe list the attachments. However, such development is currently left up to each individual e-mail client vendor as the whole e-mail foldering system is proprietary and nonstandard.

The uselessness of having separate foldering systems for documents and e-mail is further illustrated by the growing trend of users who send documents as e-mail attachments and then store them as such, i. e., never save a separate copy of the documents but instead leave them embedded in the e-mail messages. This means that users now have to search for documents in two different places. First in the traditional directories of the file system and then use their e-mail client to search for documents attached to e-mail messages. In addition, most e-mail client

UIs are not very well designed for this purpose.

In the next section we will outline a solution for unifying the document and e-mail foldering trees in the network storage and still remain compatible with most e-mail clients. This is achieved by storing messages in the network storage instead of on the PC and by providing multiple open access methods and automatic conversion between e-mail and document behind the scenes.

Note, that we do not want to force users to mix documents and e-mail messages in the same folders. However, we want to leave the choice up to the users at what level they want to mix documents and messages. On one end of the spectrum would be to have two folders at the very top level of the directory tree, one for documents and one for messages. That would be more or less the situation of today, except with unified open access. At the other end of the spectrum would be a complete mix of documents and messages in a single folder. A solution in between could look like this. A project manager, for example, could create a directory for each project. Each directory contains subdirectories, one each for project plans, technical drawings, statement of works, and one for messages. This way, if the project manager has to hand off one project to another manager, he or she can copy the entire project directory tree to the new manager, including all e-mail messages which concern the project. In today's e-mail separated world, the project manager would have to select the corresponding messages in the e-mail client and then export them and the new project manager would have to import them into his or her e-mail client.

The concept of supporting the parallel use of e-mail messages and files in a unified directory system can be extended to electronic voice mail and fax messages. Electronic voice mail and fax messages are generated by various different voice mail and fax service providers. Users typically receive these messages by e-mail, as an audio or image attachment. The PDS therefore is automatically compatible with such messages.

3.4. Access

Access to the network storage should be easy, reliable, and ubiquitous. If those conditions are not met, then users will likely not become committed to the network storage paradigm. The following three access methods seem to be the most important ones, at least in today's environment:

- Network drive.
- E-mail.
- Web.

Network drive access means the ability to install a network drive on the local PC which acts just like a local disk, except that the data is stored remotely on a network

drive. On Windows operating systems this is typically done with the “Map Network Drive...” feature of the Windows Explorer. In Linux operating systems the same task is accomplished by adding a new NFS (Network File System) mount point. This access method is important because it provides the lowest level of common file access for an operating system. It means that almost any application which reads or writes files will be compatible with the network storage, including automatic file synchronization between network storage and local temporary storage.

Web access means that users can get to their documents and messages from almost anywhere they have access to a Web browser and an Internet connection, including home, work, a hotel, airport kiosk, a TV set-top-box or game box with a built-in web browser and Internet access or a cell phone or PDA with Web browser and data connection.

E-mail access means that users can add the network storage to their e-mail client the same way they would add a new mailbox. This is important because the network storage is also a mail server and this access method lets users manage e-mail in the same way they did when e-mail was on a separate server. However, the network storage concept gives the users more choices in addition to this one as will be described later.

The following subsections describe these three access methods in more details.

3.4.1. Network Drive Access

The traditional idea of a network drive is to let users access and manage files on a remote storage disk the same way they can access and manage files on a local drive. A low-level layer of the operating system takes care of the data exchange between local workstation and remote storage and provides a consistent API to application programs. The applications therefore don’t have to deal with the networking aspect individually and most applications which were written with only local storage in mind will work with network storage without any changes. This concept is not new and most corporate users are likely using it daily. Even in home networks it starts to gain on popularity. Figure 4 shows the current Windows UI for adding a network drive to one’s PC.

The underlaying network protocol used by Windows is called SMB (Service Message Block Protocol) or in a later version CIFS (Common Internet File System). This protocol is also supported by the Linux NFS (Network File System) implementation. The PDS provides an SMB interface which makes it compatible with both, Windows and Linux based systems. There is one important distinction to ordinary network storage, however, which is e-mail.

The PDS carries both, e-mail messages and regular files. It therefore has to deal with the conversion between the two document types. In case of SMB access, all e-mail messages are presented as regular files of type `.eml` which contain the e-mail message in the standard RFC 2822 format [2]. When a user double-clicks

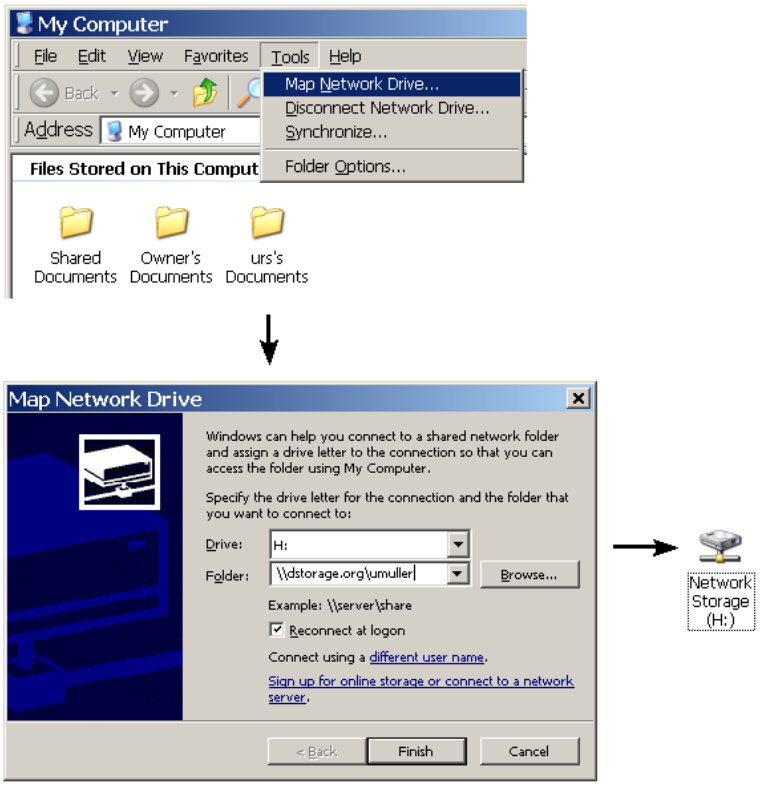

Figure 4: The current Windows UI for adding a network drive to a PC.

on a `.eml` file then the operating system launches the default application which is installed for that file type which is Outlook Express on most Windows systems and the user can read the e-mail, reply to it, forward it, or open attachments which are inside the message. The open source application Thunderbird is another popular e-mail client which supports this file format.

One further issue which needs to be addressed is the file name. E-mail messages don't have file names. The SMB therefore needs a method for creating file names when e-mail messages are displayed as regular `.eml` files. A natural method is to use a combination of the sender name and subject line with a sequence number appended, if necessary, to make the file name unique. Note, that the `.eml` files are not actually stored on the PDS. Instead, they are created on demand when a user accesses e-mail messages through the SMB interface, i. e., a network drive. Figure 5 shows an example of a directory listing which contains e-mail messages and documents. Figure 5 also shows voice mail and fax messages. The same principle is assigned to those messages. Instead of the `.eml` file format, the `.wav` format is used for voice mail messages and the `.tif` format is used for fax messages.

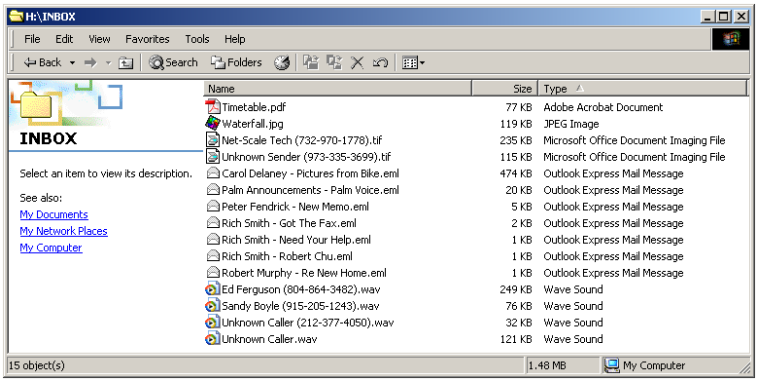

Figure 5: Listing of a directory which contains e-mail, voice mail, and fax messages in addition to regular files. The screen shot was taken from WindowsXP using the SMB protocol to connect to the PDS.

3.4.2. Web Access

Figure 6 shows a screen shot of the web interface which was created for the PDS. It shows the same directory as did Figure 5. A main advantage of the web interface is that it can be custom built for the PDS. It therefore knows the difference between e-mail, voice mail, fax, and regular files and can therefore organize the different message types on the screen. Of course, the Web UI also allows for the display of all message types in a single list, similar to the SMB interface, if that is preferred by the user.

If a user clicks on the sender name of a voice mail message the message plays instantly through the PC speakers using the default audio player which is installed on the PC. Likewise, if a user clicks on the sender name of a fax message, then the fax image is displayed on the computer screen instantly using the default TIFF viewer which is installed on the PC. Clicking on a document name opens that document and clicking on the sender name of an e-mail message opens that message. The “Details” link of voice mail messages shows additional information about the message, e. g., the forwarding trail and access to the individual message parts and spoken names. The “Call Back” link lets the user return the call using click-to-dial. This feature requires integration with a telephony server, however. The “Details” link of fax messages shows more information about the fax message, such as the forwarding trail and the “Details” link of documents shows the document’s meta information, such as title and description which can be added by the user.

3.4.3. E-Mail Access

The third supported PDS access method is through a regular e-mail client. The PDS has an IMAP4 interface. IMAP4 (Internet Message Access Protocol) is a

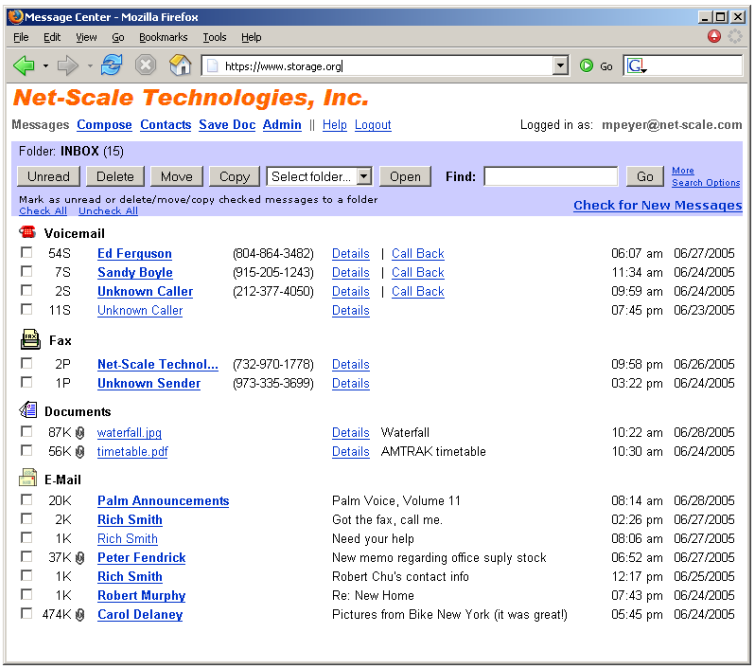

Figure 6: The web interface for listing of a directory which contains e-mail, voice mail, and fax messages in addition to regular files.

standard mail access protocol [3] which is supported by most e-mail clients, including Outlook Express, Outlook, Thunderbird, and Eudora. Figure 7 shows the same directory shown in Figure 6 but accessed through a regular e-mail client (Outlook 2003 in this case).

In this case we have the opposite problem we had for the network drive access, namely we need a method for converting regular files into an e-mail message such that they can be displayed and accessed by a regular e-mail client. This is achieved by wrapping the document into a small e-mail envelope. The subject of that e-mail envelope contains the file name and the message body contains the meta data. An example of a picture file is shown in Figure 7.

3.5. Document Sharing

Sharing a document with a recipient can theoretically be done in a number of ways:

1. Physically attach the document to an e-mail message.
2. Make the document publicly available and send a pointer to the recipient and let the recipient retrieve the document at a later time.

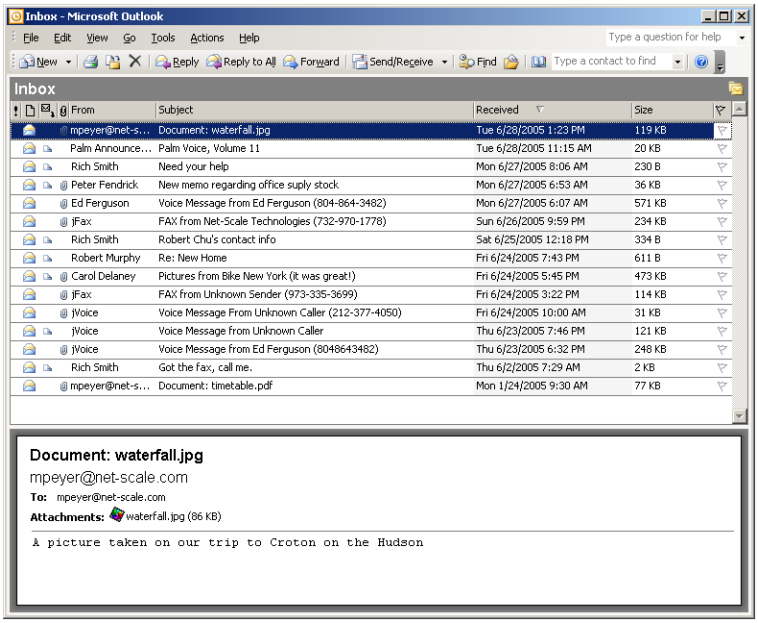

Figure 7: The e-mail interface for listing of a directory which contains e-mail, voice mail, and fax messages in addition to regular files. The screen shot was taken using Outlook 2003.

3. The same with the addition of access control to make the document accessible only to the recipient and not the general public.

Number 3 is probably the preferred solution in most cases. It prevents recipients' mailboxes from being overfilled with large documents, recipients don't have to store a copy of the document, they can just store the link to it, the sender can modify the document and the recipient will always have access the latest version of it. Finally, if the recipient accidentally forwards the message containing the document link to a person or group who should not have access to that document, then access is automatically denied through the access control mechanism. In other words, if the recipient of a document link forwards that link to somebody else, then that person does not automatically gain access to the document. Access remains solely under control of the original sender of the document.

Unfortunately method 3 is also the most complicated one. It involves publishing the document on a suitable server, installing access control, opening access for the recipient, send the actual e-mail with document link to the recipient and also sending the access credentials to the recipient, e. g., in a separate e-mail message. This requires an effort which goes far beyond what most users are willing to do. The PDS has a nice automatic solution to this problem.

Using the web interface, a user composes the message to the recipient. Instead of choosing "attach document" the user chooses "include link to document". This selects the difference between attaching a document to the e-mail message

and sending a link to the document. By doing so, the system automatically opens access for the recipient's e-mail ID to the document. The document itself does not need to be published as it is already stored in the network and can be made accessible through the existing web interface. If this was the first link sent to this particular recipient, then the system also creates a second automatically generated e-mail message to the recipient which contains the access credentials required by the recipient to retrieve the document.

When the recipient clicks on the document link he or she will be redirected to the web interface of the PDS. It will ask for the recipient's e-mail ID and a password. The credentials will be stored in a cookie on the recipient's browser upon the user's request so the login procedure has to be performed only once. Note, that this procedure can be done using nonsecure or secure e-mail, depending on the security requirements of a particular user group.

3.6. Document Pointer Indirection

Most readers will be familiar with DNS (Domain Name System). DNS allows users to remember meaningful names instead of IP addresses. DNS does more, however. It allows an owner to move a particular web site from one IP address to another without the visitors of the web site noticing. This is the case, for example, when a company changes its web hosting service provider. Furthermore, DNS also allows for load balancing by listing multiple IP addresses for a single DNS name.

It would be nice to make these same advantages available for personal document storage, including allowing users to move their documents from one server to another without making all the links pointing to the documents invalid. Unfortunately, DNS is not well suited to be applied to documents. For one, it cannot store document pointers, only IP addresses, and it provides no access control. There is a new system, however, which is perfectly well suited for this purpose. It is called the *Handle System* [4]. The Handle system works similar to DNS but it does provide a small database table for each entry which can hold arbitrary types of data instead of just IP addresses and it provides ownership and access control. Its integration with the data storage is straight forward as shown in Figure 8.

4. Conclusions

This paper started by exploring the inadequacy of personal document storage on a PC or Laptop drive as it is customarily done by most users today. It further emphasizes that the situation will likely become worse in the near future with the amount of personal electronic data such as pictures being created by users rapidly growing. The paper continues with outlining a complete solution based on network storage. The solution is designed to be open and avoids proprietary technology such that no limits are imposed on independent vendors and service providers to extend upon

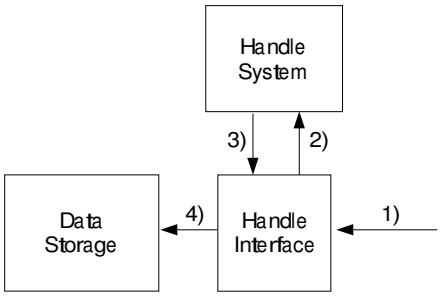

Figure 8: The Handle System integration. When a new document is sent to the data storage 1) then the Handle Interface automatically requests the generation of a new Handle 2). The new Handle is sent back to the interface 3) and both, document and Handle are stored in the Data Storage 4).

and improve the solution. The solution also provides plentiful ways for end users to access and interact with the data storage without requiring the installation of any special clients. The existing file browsers, network interfaces, e-mail clients, and web browsers of today's PCs are used for this purpose. Furthermore, the solution provides a way for unifying e-mail and document foldering systems which are so unnaturally separated on in most PC environments today. Finally, the solution includes a pointer indirection method for document access using the open Handle system. This allows users to move their documents between service providers without loosing any existing pointers to their documents.

References

[1] Katie Hafner. Even digital memories can fade. *New York Times*, November 10 2004. Available on-line: `http://www.nytimes.com/2004/11/10/technology/10archive.html?ex=1101060083&ei=1&en=80f7ed6e41d62ae1`.

[2] Peter W. Resnick. Internet message format. Standards Track Request for Comments: RFC 2822, QUALCOMM Incorporated, April 2001.

[3] Mark R. Crispin. Internet message access protocol - version 4rev1. Standards Track Request for Comments: RFC 2060, Networks and Distributed Computing, University of Washington, December 1996.

[4] Sam X Sun, Larry Lannom, and Brian Boesch. Handle system overview. Standards Track Request for Comments: RFC 3650, Corporation for National Research Initiatives (CNRI), November 2003.