8-15-1997

Web-Based Tools for Team-Based Development

Judith Barlow
American University, jbarlow@american.edu

Web-Based Tools for Team-Based Development

Judith Barlow

Computer Science and Information Systems Department
American University
4400 Massachusetts Avenue, NW
Washington, DC 20016-8116
email: jbarlow@american.edu

"*Internet tools*" such as web browsers (e.g. Netscape Navigator), video conferencing software (e.g. Enhanced CU-SeeMe), remote login software (e.g. telnet) and file transfer programs (e.g. ftp, Fetch) are designed to provide the same "look and feel" regardless of hardware platform or operating system. They can provide support for team-based development by collaborators working in close physical proximity as well as those separated by large physical distances. Web browsers can be used to access electronic documents on remote and local computers while providing an effective medium for passing documents between incompatible applications (e.g. Macintosh and Windows application software) and providing access to files on machines which are otherwise inaccessible to fellow collaborators.

Much of the success of these "web-based tools" is due to the ease in which users working in incompatible computing environments can share a wide variety of complex file formats. Text, graphics, video, audio, and word processing files can be shared, often without regard for an individual collaborator's development environment. The machine-independent nature of such web-based tools makes them especially appealing to organizations with heterogeneous internal computer networks.

We focus on the use of web browsers for supporting collaboration across the Internet, over an Intranet, and between user directories on a common host computer. In all cases, the same characteristics which make "Internet tools" especially appealing can pose some potential security problems. Administrators of web-accessible systems must take extra precautions to protect their systems from the activities of external users. Supervisors of web-based collaborative activities must also concern themselves with version management of files and applications related to new releases of shared software tools.

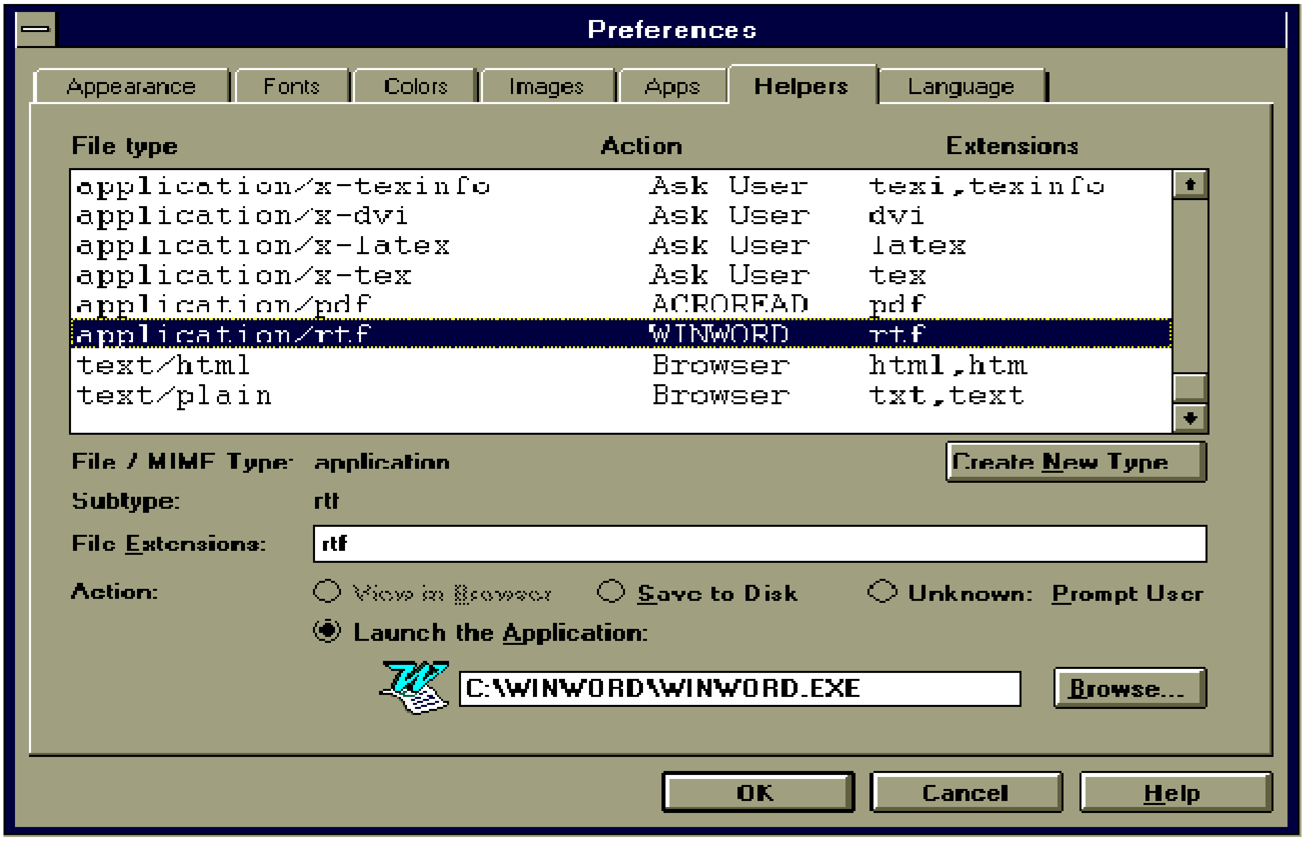
File Sharing: "*Incompatible*" *Files*

Web browsers can be used to provide easy access to otherwise incompatible files as well as providing a means of accessing files which are otherwise inaccessible. Web browsers allow collaborators to easily share files with little concern about a team member's hardware platform or operating system. For example, a collaborator using Microsoft Word (or WordPerfect) on a Macintosh can use her web browser to "download" a Microsoft Word (or WordPerfect) document created in Windows 95 (or Windows 3.1 or Windows NT) from the web and edit the document on her Macintosh. She can save her edited document in an accessible folder which then allows collaborators using PCs or UNIX workstations to access and manipulate the file in their own computing environments.

Even collaborators using different vendor's application software (e.g. WordPerfect, Microsoft Works) can use web browsers for effective file sharing. Most word processing software allows files to be saved in alternative formats. Rich Text (Interchange) Format files can be manipulated by most word processors. Sharing these documents is as simple as configuring a "Helper" as shown in Figure 1.

Collaborators still need to concern themselves when upgrading applications whose files are shared by others. Many applications (e.g. Adobe Acrobat, Apple QuickTime, Microsoft PowerPoint) provide free viewers which can be configured for users' web browsers; however viewers do not provide editing capabilities, so are useful only for one-way exchange of information.

Most application software provides backward editing compatibility; however, many do not allow owners of older versions to view or edit documents created in newer versions of the same software. This can be especially troublesome as release dates for software generally vary for different operating systems. For example, Microsoft Office documents created in Office97 are not viewable or modifiable by users of older versions of Microsoft Office. Since Office97 is not yet available for Macintosh or Windows 3.1 computers, users of Office97 who want to take advantage of its new, enhanced features can, for now, only collaborate with Windows 95 and Windows NT users.



File type	Action	Extensions
application/x-texinfo	Ask User	texi,texinfo
application/x-dvi	Ask User	dvi
application/x-latex	Ask User	latex
application/x-tex	Ask User	tex
application/pdf	ACROREAD	pdf
application/rtf	WINWORD	rtf
text/html	Browser	html,htm
text/plain	Browser	txt,text

File / MIME Type: application

Subtype: rtf

File Extensions: rtf

Action: ○ View in Browser ○ Save to Disk ○ Unknown: Prompt User

● Launch the Application:

C:\WINWORD\WINWORD.EXE

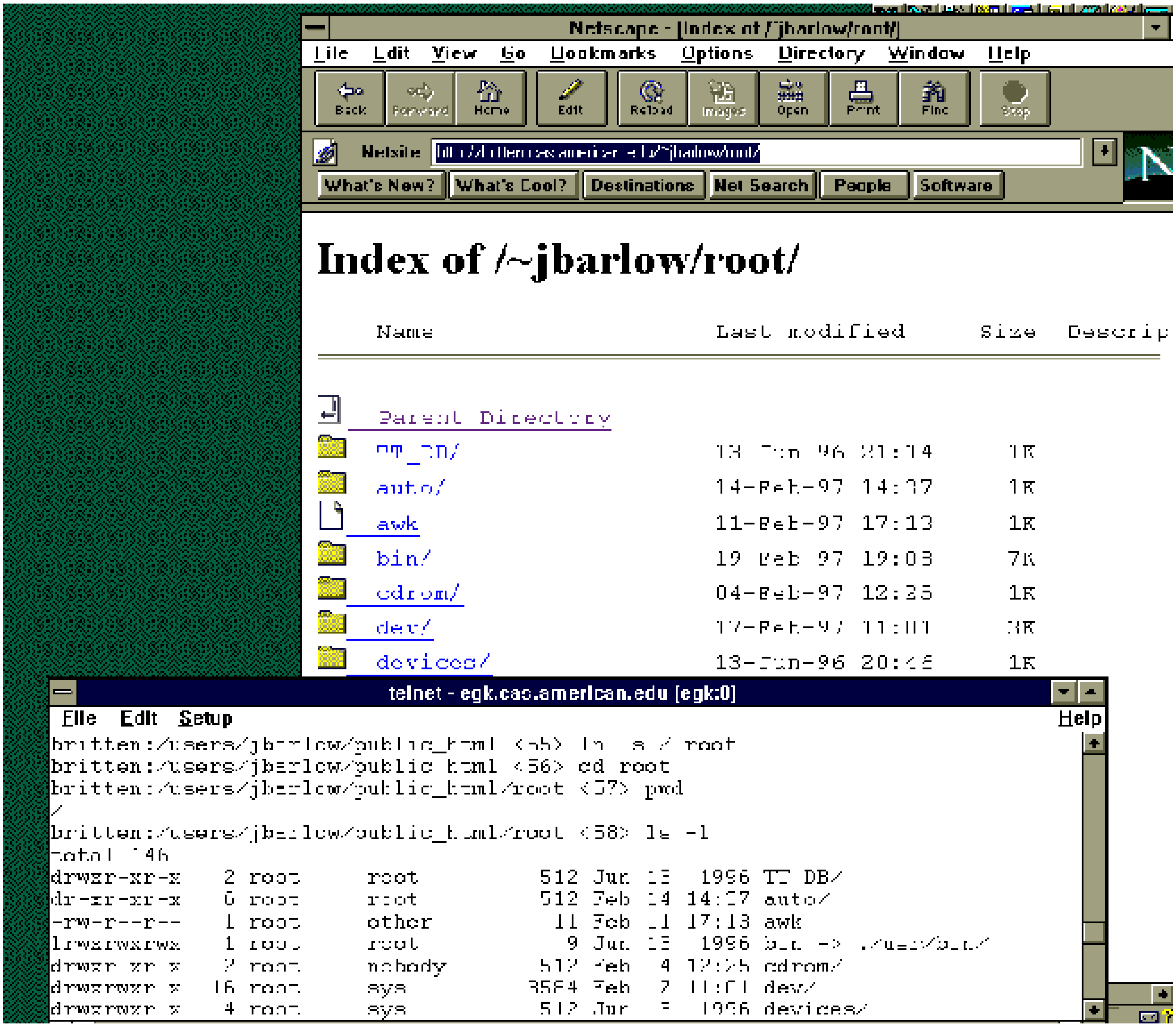
Figure 1: A Netscape Navigator ***Helper*** for **rtf** (rich text format) files if defined as **WINWORD.EXE**, the desired word processor's executable application program. Now every time that a link to a rich text format file is selected, the selected application software, in this case Microsoft Word, will be run to view the file Users have access to the word processor's full functionality; however they may experience problems if the rich text format file was created using a newer version of Word.

File Sharing: "*Inaccessible*" Files

Another aspect of file sharing using web browsers is that of providing access to remote or otherwise inaccessible directory structures. For security or other pragmatic reasons, collaborating team members may not have access privileges to a common multi-user computing environment. Even in cases where all team members have access to a common host (e.g. mainframe for electronic mail), the common host may not be the users' development environments. The common host may not easily serve as a platform for exchanging complex file types or be a desirable computing environment for all collaborating team members.

Web browsers can be used in conjunction with file access privileges to grant access to directory structures which would otherwise be available only through access modes provided by system administrators. The owner of a filestore on a system accessible to the web browser need only create an ***alias*** or ***symbolic link*** to other files on the system. The ability of an individual user to "bypass" system administrators and grant access to system files is demonstrated in Figure 2.

A user has created a symbolic link to the system root directory. Now any user with a web browser can traverse the UNIX system's directory structure. The effect is demonstrated in the Netscape window where all files accessible to general system users are now accessible to **all** external (unknown) users. Now anyone with web access can browse the system directory hierarchy with the same privileges to the least powerful valid system user. While there is no real concern about data loss or file corruption, the ability to inadvertently provide public access to system files may be of concern.



Index of /~jbarlow/root/

```
    Name                       Last modified       Size  Descrip
--------------------------------------------------------------------

    Parent Directory
    TT_DB/                     13-Jun-96 21:14      1K
    auto/                      14-Feb-97 14:37      1K
    awk                        11-Feb-97 17:13      1K
    bin/                       19-Feb-97 19:03      7K
    cdrom/                     04-Feb-97 12:25      1K
    dev/                       17-Feb-97 11:01      3K
    devices/                   13-Jun-96 20:45      1K
```

```
britten:/users/jbarlow/public_html <55> ln -s / root
britten:/users/jbarlow/public_html <56> cd root
britten:/users/jbarlow/public_html/root <57> pwd
/
britten:/users/jbarlow/public_html/root <58> ls -l
total 146
drwxr-xr-x   2 root     root         512 Jun 13  1996 TT_DB/
dr-xr-xr-x   6 root     root         512 Feb 14 14:37 auto/
-rw-r--r--   1 root     other         11 Feb 11 17:13 awk
lrwxrwxrwx   1 root     root           9 Jun 13  1996 bin -> ./usr/bin/
drwxr-xr-x   2 root     nobody       512 Feb  4 12:25 cdrom/
drwxrwxr-x  16 root     sys         3584 Feb  7 11:01 dev/
drwxrwxr-x   4 root     sys          512 Jun  3  1996 devices/
```

Figure 2: A user of the UNIX system has created a symbolic link to the system's root directory. The directory **~jbarlow/root** in the Netscape window is the directory **/users/jbarlow/public_html/root** in the telnet window which is actually the **system root directory**!

Coordinating Security and Access

Developers of web server software are concerned with security issues and the need to limit access to system resources. A common strategy is to isolate all web-accessible directories, folders, and files from other system files; but as demonstrated in Figure 2, a single user can inadvertently sabotage these simple security efforts.

Team managers need also concern themselves with the task of coordinating software upgrades or devising a protocol that assures that all collaborators, regardless or applications software version, can manipulate as well as access information.

Summary

Although Internet tools such as web browsers can provide a simple means of sharing files which would otherwise not be easily accessible, web server software is not yet mature enough to provide an environment in which files are secure. The same characteristics of web browsers which has made them successful pose potential security risks for those who provide access to information on file systems which also include secure datafiles or directories. Database designers learned long ago that it takes more than a secret schema to keep data secure.

References available upon request from author.