

From: timbl@dxcern.cern.ch (Tim Berners-Lee)
From: timbl@dxcern.cern.ch (Tim Berners-Lee)
To: dimou@dxcern.cern.ch, rtb@dxcern.cern.ch
Subject: Doc Int Pos at Cern
Date: Thu, 29 Apr 1993 23:14:38 +0200

Where does WWW fit into your paper?

Well, it fits in naturally into section 3, before or after FTP. The WWW story in fact picks up many of the points made throughout the paper, so in a way the paper provides a backdrop to the WWW project, as it explains clearly some of the problems which spurred WWW's development.

First of all, WWW document transfer is initiated by the receiver. There is an assumption throughout that document transfer is initiated by the sender, and so in the FTP chapter, you write "The most commonly used technique is therefore for the sender to store the document... and then inform the reader... requesting him to copy the document".

This is a lop-sided view. Communication takes two, and whilst it is necessary to be able to notify people with unsolicited messages, also there is unprovoked enquiry on the part of readers. The use of mail and news for notification, and retrieval protocols for bulk access will do a lot to unclutter the networks, our disks and our lives.

WWW is basically like FTP -- client server, receiver initiated. Here are the differences.

The clients are easier to use. The two operations of selecting from menu (hypertext document) and or typing text for a search are all you need to master. No ftp client commands, no messages like 200 PORT command OK.

The WWW clients can in fact access any FTP server. So you can use an FTP server if you have one. You might want to write some hypertext to be more informative, more guiding, to your readers than a raw directory listing, but if all you have time to do is name a directory, fair enough.

The advantages of running a HTTP (w3's own protocol) server instead of an FTP server are:

1. It's faster. Only one connection, one request/response delay, compared with two connections and maybe five request/response delays for FTP. Speed makes a LOT of difference to a user's effectiveness in getting knowledge from the information base. Research showed effectiveness slows after 100ms response time.
2. It makes less load on your CPU. The connections are not held open, freeing resources on your computer. The protocol is stateless, so server processes exit immediately, saving space and time. HTTP server load is negligible even on our server with 7k requests per day.
3. The server has some nice features. a. It will make a nice job of displaying your directories if you have README files in them: it can automatically put the README at the top or

bottom of the directory listing.

b. It allows you to map document names into filenames with an extra very flexible mapping level so that the coordinates of your documents won't change when you shuffle the disks around.

c. It can log each request, for feedback on which IP addresses are reading what. (some ftp daemons do this too).

4. The HTTP protocol knows about the difference between text and binary files, you don't have to.

5. The HTTP protocol knows about the difference between postscript and text. You can put 6 versions of the same document in different formats on your disk. The server at request time will figure out which one to send. For example, you can put a postscript and a plain text version, .ps and .txt. If you refer to the document as .multi, then the user gets postscript if he can handle it, else plain text. (the client is configurable to specify what formats are acceptable, and whether they should be displayed or given to another application to display. The client sends the list of formats to the server, along with numerical weights to say which it prefers. The server makes a balanced judgement based on its own tables. In case you wanted to know how it worked. The job of configuring a multimedia WWW client is equivalent to the job of configuring a MIME user agent.) None of this of course is seen by the user.

There are other things I could go into, but for a summary that is enough.

OTHER POINTS ON THE PAPER

2: document types. First bullet.

"dissemination of information [sic] such as printable documents sent out for reference

should be in non revisable form such as postscript."

I think should is too strong. There are some who are paranoid about people making unauthorised changes, but that is going too far.

Advantages of sending out source include people's ability to search through it, for example, and view it in more convenient ways.

2.1 Postscript fonts. The IETF I think wisely suggest sticking to Courier, Helvetica, and Times for anything which you want to distribute. I know from experience that nice though Palatino is, you have to forget it!

2.3 Plain text. Character sets. Some mention of ISO latin 1 might be in order as a standard the Scandinavians push for with some success. (sorry, Maria, about the greek)

2.4 If this is intended as guidance, you might mention that (on the PC at least), Saving As "text with layout" is fairly useless but printing to a "generic text/only" printer if you have the driver is much more useful.

2.5 You don't mention the DTD differences. No matter. As a separate paragraph you could mention HTML in passing which is a very simple DTD, going to be an RFC, and which every W3 viewer can intrinsically understand. While simple, it is sufficient for general online documentation, overviews, etc, and specifically includes ISO Latin1

support. (Some terminals have to approximate, of course!)

2.7.2 This is all very interesting, and stuff we will all get into more and more as multimedia WWW support hits us. Arthur Secret (room 007) is looking at various graphics formats for scanned mages, finding scanned text is best in compressed GIF but pictures much better in JPEG, etc. Check out a list of viewers in the vatican culture online exhibit. (Under www subjects, under literature & art). If yu could assemble or include or point to lists of info about formats, and public domain conversion software which we could bolt on to servers, that would be very useful indeed.

3.3 Many comments about conversion from mac or pc to unix line break characters. Of course it is fairly easy to make a w3 server to do this on the fly, so that directories which are in fact on foreign machines are served to the world through a filter.

RECOMMENDATIONS

Seems to me that group servers should run www servers. (of course :-) Someone in each group should think a little about document names before just using filenames, as document anmes live a long time. Remember to make an alias for the server machine as that will have to live a long time too.

AUFS looks like a good way for people to drop things for publication into a public server which is w3-visible. The server can be made to recognize the files and convert them on the fly for distribution.

Things which we will need which we haven't got include a way of taking mail and automatically filing it according to date, subject, sender, which dummy address it was sent to, etc. Many people need this, includig me. I have a contributed program which is supposed to do it, haven't had time to try it.

A utility to 'find' any files added during the last week and mail off a [hypertext] list of them to interested parties is another useful gadget ... probably a one-liner.

Note there is a W3 main server which allows any w3 document to be requested by mail. Given this and a robot filer, people who are not on internet or deernet but do have mail can have a reasonable chance of participating in the work of a group.

Hope this helps..... keep up the good work.

Tim

PS: I just splurged this down a modem from Nance's PC so if there are characters missing, I aplogize... if someone has a book with the modem command set for turning on error correction I could use it....