The Internet: a tutorial

by J. Crowcroft

The Internet is a world-wide packet-switched network that connects together well over 10 million computers in over 100 countries for the purpose of information sharing. This paper is a tutorial on Internet technology — how it works now and how it is changing as the network becomes more commercial. It covers the underlying design of the Internet, the 'connectionless service model', and the applications that run over the network: the World Wide Web and other popular systems that have made the network growth so explosive in recent years.

1 The Internet: What kind of service?

The Internet is the Information Superhighway; it has become a cliché to say so. However, before embarking on a drive around the World wide Web (WWW), it is important to understand how the roads themselves work (and to understand who pays road tax).

The Internet is undergoing a stormy adolescence as it moves from being a playground for academics to a commercial service. With more than 50% of the network commercially provided and more then 50% of the subscribers being businesses, the Internet is now a very different place from what it was in the 1980s. Growth has occurred most sharply in Europe and in the commercial sector in the last two years.

It is normal to describe the Internet under two headings. The first is host software (applications and support); this forms what one might call the 'information services'. The second is network support; this is the access and transfer technology used to move the information around.

Hosts, networks and routers

The components that make up the Internet, illustrated in Fig. 1, are threefold:

(*a*) *Hosts:* These are the workstations, PCs, servers, and mainframes on which applications are run.

(*b*) *Networks:* These may be local-area nets (Ethernet LANs for example), point-to-point leased lines or dial-up (telephone, ISDN, X25) links that carry traffic between one computer and another, or wide-area networks (WANs) using switching technology such as SMDS (Switched Multimegabit Data Service) or ATM (asynchronous transfer mode).

(*c*) *Routers:* These glue together all the different network technologies to provide a ubiquitous service to deliver packets (a packet is a small unit of data convenient for computers to bundle up data for sending and receiving). Routers are usually just special purpose computers which are good at talking to network links. Some people use general purpose machines as low-performance (low-cost) routers, e.g. PCs or Unix boxes with multiple LAN cards or serial line cards or modems.

Names, addresses and routes

Every computer (host or router) in a well run part of the Internet has a name. The name is usually given to a device by its owner. Internet names are actually hierarchical and look rather like postal addresses. For example, the name of my computer is waffle.cs.ucl.ac,uk: I allocated it the name *waffle*; the department in which I work called itself CS (Computer Science); the university it is in called itself UCL (University College London); the academic community called themselves ac; and the Americans called us the UK. The name tells me what something is *organisationally.* The Internet calls this the Domain Name System.

Everything in any part of the Internet that is to be reached must have an address. The address tells the computer in the Internet (hosts and routers) where something is topologically. Thus the address is hierarchical. My computer's address is 128.16.8.88. My university asked the IANA (Internet Assigned Numbers

Principal abbreviations

DNS	=	Domain Name Service
FTP	=	File Transfer Protocol
GIF	=	Graphics Interchange Format
HTML	=	HyperText Markup Language
HTTP	=	HyperText Transfer Protocol
IANA	=	Internet Assigned Numbers Authority
IP	=	Internet Protocol
LAN	=	local-area network
MIME	=	Multipurpose Internet Mail Extensions
NIC	=	Network Information Center
RDP	=	Reliable Data Protocol
RFC	=	Request for Comments
SMTP	=	Simple Message Transfer Protocol
TCP	=	Transmission Control Protocol
UDP	=	User Datagram Protocol
WWW	=	World Wide Web

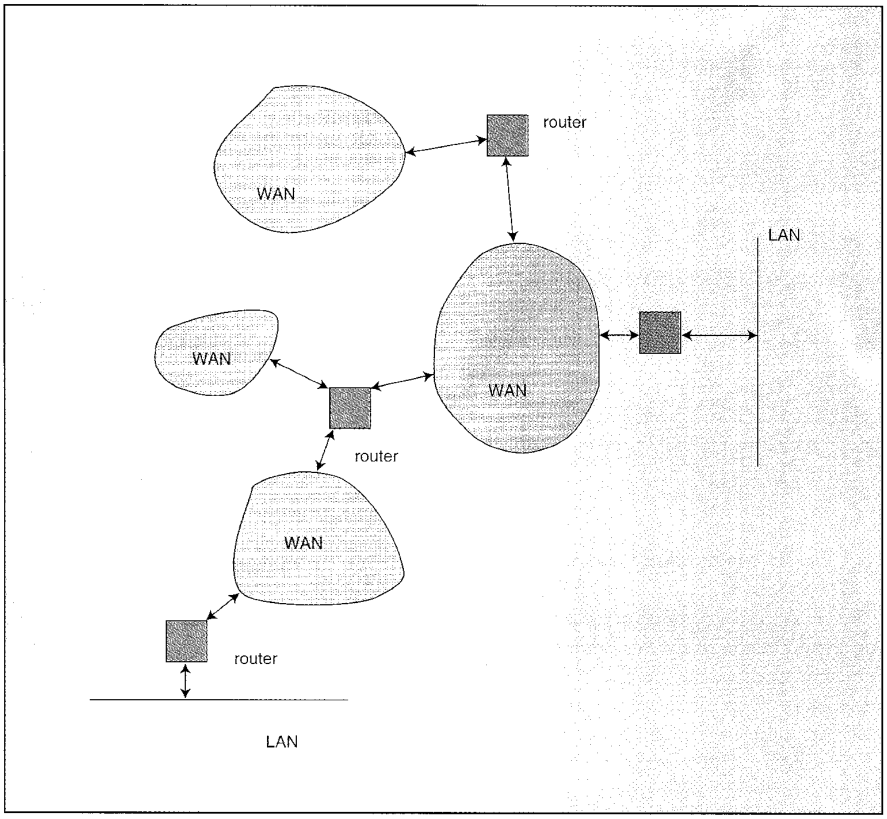


Fig. 1 A piece of the Internet

Authority) for a network number* and was given the number 128.16.*x*.*y* in which we could fill in the *x* and *y* how we liked, to number the computers on our network. We divided our computers into groups on different LAN segments and numbered the segments 1–256 (*x*), and then the hosts 1–256 (*y*) on each segment. When an organisation asks for a number for its net, it is asked how many computers it has and is assigned a network number big enough to accommodate that number of computers. Nowadays, if you have a large network, you will be given a number of numbers!

Everything in the Internet must be reachable. The route to a host will traverse one or more networks. The easiest way to picture a route is by thinking how a letter to a friend in a foreign country gets there.

You post the letter in a postbox. It is picked up by a postman (LAN) and taken to a sorting office (router). There, the sorter looks at the address and sees that the letter is for another country, and sends it to the sorting office for international mail. This then carries out a similar procedure. And so on, until the letter gets to its destination. If the letter was for the same 'network' then it would get immediate local delivery. Notice that all the routers (sorting offices) don't have to know all the details about everywhere, just about the next hop to go to. Notice also that the routers (sorting offices) have to consult tables of where to go next (e.g. international sorting office). Routers chatter to each other all the time figuring out the best (or even just usable) routes to places.

The way to picture this is to imagine a road system with a person standing at every intersection who is working for the Road Observance Brigade. This person (Rob) reads the road names of the roads meeting at the intersection and writes them down on a card, with the number 0 after each name. Every few minutes, Rob holds up the card to any neighbour standing down the road at the next intersection. If they are doing the same, Rob writes down their list of names, but adds 1 to the numbers read off the other card. After a while, Rob is now telling people about the neighbour's roads several roads away! Of course, Rob might get two ways to get somewhere! Then he crosses out the one with the larger number.

Performance

The Internet today moves packets around without due regard to any special priorities. The speed at which a packet goes once it starts to be transmitted is the speed of the wire (LAN, point-to-point link, dial-up link etc.) on the next hop. The range of communication technology speeds is illustrated in Fig. 2. However, if there are a lot of users, packets get held up inside routers (like letters in sorting offices at Christmas). Because the Internet is designed to be interactive, unlike the mail (even electronic mail) system with its slow turnaround, routers generally do not hang on to packets for very long. Instead, they just 'drop them on the floor' when things get busy! This then means that hosts have to deal with a network that loses packets.

* The task of allocating numbers to sites in the Internet has now become so vast that it has been delegated to a number of organisations around the world. Ask your Internet provider where they get the numbers from if you are interested.

Hosts generally have conversations that last a little longer than a single packet — at least one packet in each direction, but usually several in each direction.

In fact, it is worse than that. The network can automatically decide to change the routes it is using because of a problem somewhere. Then it is possible for a new route to appear that is better. Suddenly, all packets will follow the new route. But if there were already some packets half way along the old route, they may get to their destination after some of the later packets (a bit like people driving to a party and some smart late driver taking a short cut and overtaking the earlier leavers). So a host has to be prepared to put up with out of order packets, as well as lost packets.

Protocols

All this communication is done using standard 'languages' to exchange blocks of data in packets, simply by putting 'envelopes' or wrappers called 'headers' around the packet.

The work of routing and addressing is done by the Internet Protocol, or IP. The work of host communication is done by the Transmission Control Protocol, or TCP. TCP/IP is often used as the name for the Internet protocols, including some of the higher level information services. TCP does all the work to solve the problems of packet loss, corruption and reordering that the IP layer may have introduced, through a number of *end to end* reliability and error recovery mechanisms. If you like, you can think of IP as a bucket brigade and TCP as a drainpipe.

So to send a block of data, a TCP header is added to it to protect it from wayward network events and an IP header to get it routed to the right place.

Host and applications

The emergence of some simple APIs (application programming interfaces) and GUIs (graphical user interfaces) has led to the rapid growth of new user-friendly applications in the Internet. Information services provided by archive and Web servers are accessible through WWW and Mosaic, Archie and Prospero, gopher, and WAIS and Z39.50

Fig. 3 shows a 'screendump' of a workstation running Mosaic, a popular client program for accessing the Web servers around the world. In this picture are a window showing a map of the UK and two other pictures, both GIFs (graphics interchange formats) or photographs. The top right picture is of University College London. The lower right picture is one of the satellite images stored on Edinburgh University Computing Service's Web server every hour from a live satellite weather feed, which comes from one of the many weather satellites that send out unencrypted images periodically over the ether. Many sites simply point a satellite dish in the right direction and soak up the data for later dissemination on the terrestrial Internet.

The value of the service is now clearly becoming the value of the information rather than the communication channel. This means that some mechanism for charging and auditing access to information is a new requirement. This must be a secure mechanism to assure people that charges (or audit trails) are correct.

Fig. 4 shows a screendump with three applications running. The top left is a Gopher Client, the top right is Archie, and the bottom right is WAIS.

From all of these interfaces, having found a piece of information, we can retrieve it, print it or mail it to other people.

2 Information servers — Are you being served?

We are familiar with ways to get information in the non-

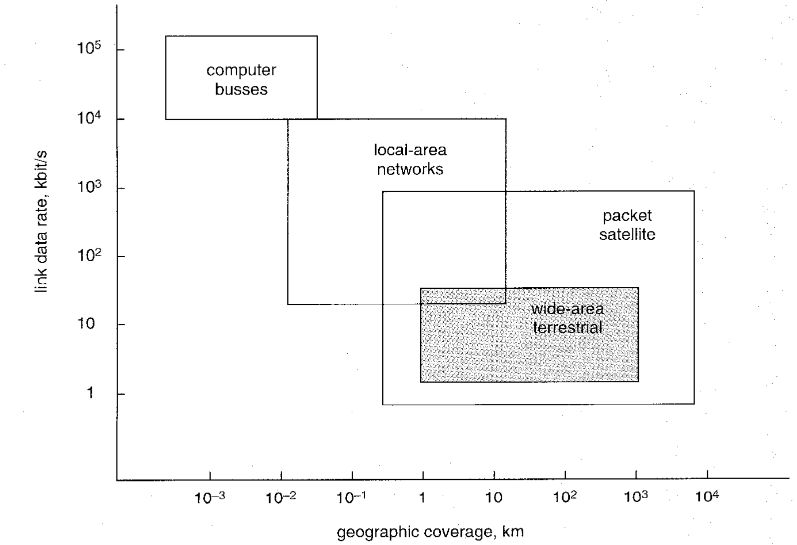
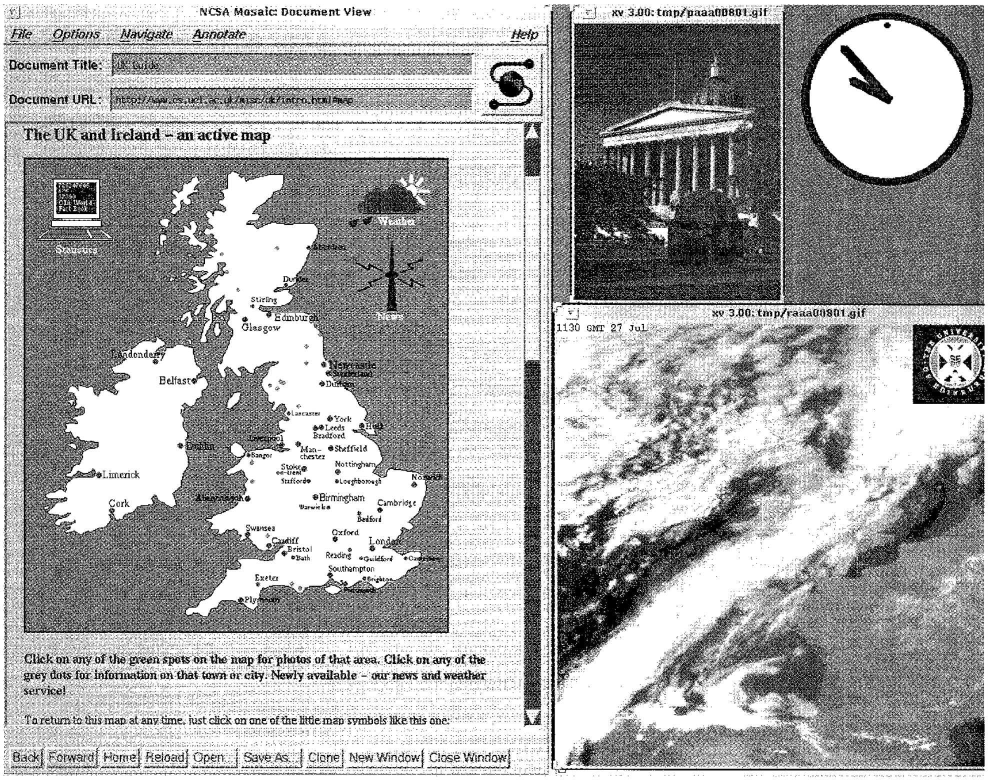


Fig. 2 Range of network performance



NCSA Mosaic: Document View

Document Title: UK Guide

Document URL: http://www.cs.ucl.ac.uk/misc/uk/intro.html#map

The UK and Ireland – an active map

Click on any of the green spots on the map for photos of that area. Click on any of the grey dots for information on that town or city. Newly available – our news and weather service!

To return to this map at any time, just click on one of the little map symbols like this one:

xv 3.00: tmp/paaa00801.gif

xv 3.00: tmp/raaa00801.gif

1130 GMT 27 Jul

Fig. 3 'Screendump' of workstation running Mosaic

network world: we can go to a library, or buy a book in a bookshop; we can telephone companies or individuals by looking up their names in a phonebook; we can sit around and watch TV or listen to the radio. In the networked world, there are a number of ways of carrying out the same kind of activities.

Different kinds of information services have different models of use and different ways of holding information. Almost all fit into the 'client/server' model that has become widespread in distributed computing. Client/server communication is quite easy to understand in terms of roles and is very closely analogous to what happens in a shopping situation. An assistant in a shop awaits a customer. The assistant doesn't know in advance which customer might arrive (or even how many — the store manager is supposed to make sure that enough assistants are employed to just about cope with the maximum number of shoppers arriving at any one time). A server on the network is typically a dedicated computer that runs a program called the server. This awaits requests from the network, according to some specified protocol, and serves them, one or more at a time, with regard for who they come from.

There are a variety of refinements of this model, such as requiring authentication or registration with the server before other kinds of transactions can be undertaken, but almost all the basic systems on the Internet work like this for now.

Information servers can be categorised along a number of different axes:

- *Synchronous versus asynchronous:* Synchronous servers respond as you type/click at your computer. Asynchronous ones save up their answers and return them some time later. Sometimes, your system will actually not even send the request for information to the server until you have finished composing the whole request, or even later, to save time (and possibly money, since night-time network access may be cheaper and/or faster).
- *Browsable versus searchable:* Browsable servers allow you to move from one piece of information to another. Typically, the managers/keepers have structured the information with links, or else the information is hierarchical (like most organisations' job structures or payrolls). Searchable servers allow you to search for particular items by giving keywords. Usually, this means that the managers of the information have created indexes, although sometimes it just means that the server is running on a very fast computer that can search all through the data. This latter approach is becoming increasingly impossible as the quantity of information kept online grows to massive proportions.
- *Distributed versus replicated:* Distributed information servers hold only the information entered at their site and may have links for the user to follow to other servers at other sites. Replicated systems copy the information around at quiet times, so that all servers

are replicas of each other. This means that it doesn't matter which server you access in the latter case, so you might as well go for the nearest or cheapest (likely to be both).

Transport protocols

The Internet provides a way to get packets (convenient units of data for computers and routers) from any host computer to one or more other host computers. However, the network protocols make no guarantees about delivering a packet. In fact, a packet may get lost, may arrive after others sent later or may be distorted. A packet might even arrive that simply wasn't sent!

To counter this, host computers incorporate transport protocols, which use the Internet to carry the application information around but which also send a variety of other information to provide checking and correction or recovery from such errors.

There is a spectrum of complexity in transport protocols, depending on the application requirements. Three representative ones are:

(*a*) *The User Datagram Protocol (UDP):* UDP is a 'send and forget' protocol. It provides just enough information at the start of each packet to tell what application is running and to check if the packet was distorted on route. UDP is used by applications that have no requirement for an answer, typically, and don't really care if the other end received the message. A typical example of this might be a server that announces the time on the network, unsolicited.

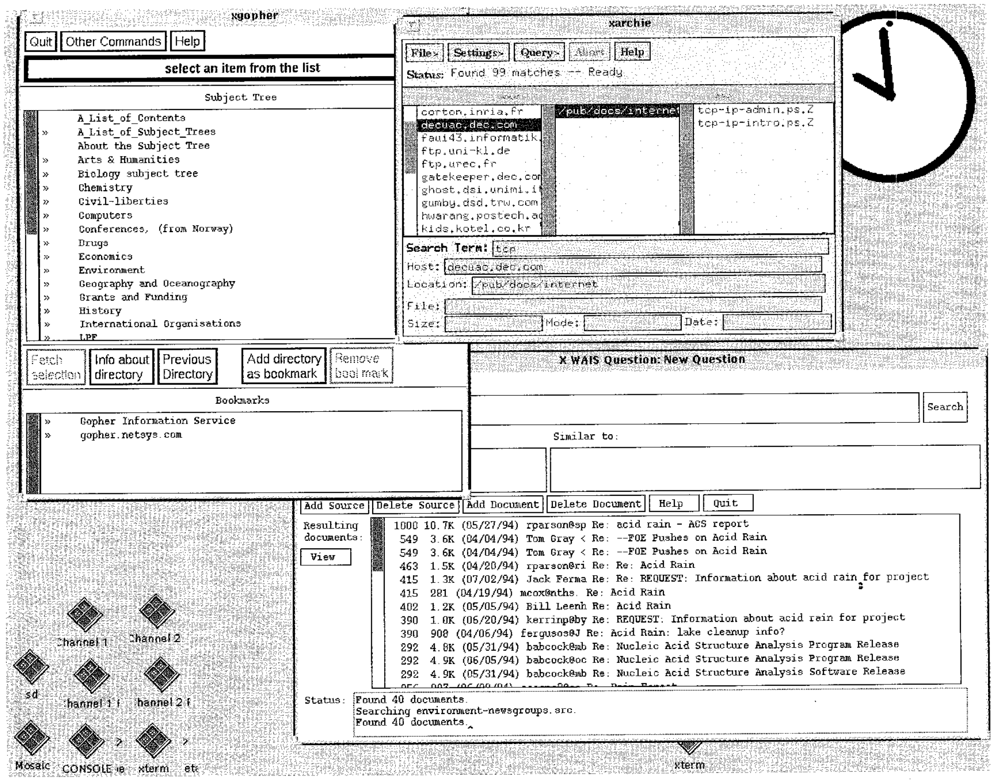
(*b*) *The Reliable Data Protocol (RDP):* RDP is a generic name for a collection of protocols — the most relevant one here is that used by Prospero. RDP type protocols are similar to TCP (see below), but with reduced complexity at the start and end of a conversation, and with good support for sequences of exchanges of chunks of data, often known as 'remote procedure calls' and sometimes incorrectly called 'transactions'. The term 'transaction' refers to an atomic, or uninterruptible exchange of data and is usually much more expensive to implement than a simple remote procedure call.

(*c*) *The Transmission Control Protocol (TCP):* TCP is the protocol module that provides reliability and safety. It is designed to cope with the whole gamut of network failures and adapts elegantly to the available resources in the network. It even tries to be fair to all users.

File Transfer Protocol (FTP)

The original information service was called FTP. This is probably the world's least friendly information service. It operates really at the level of machine–machine communication and is used by more modern client programs just as a way of getting something from a server. However, it is still used as a sort of lowest common denominator means of access to a file on a remote computer.

Internet FTP is interactive or synchronous, which means that you formulate your commands as you type at the terminal. FTP maintains a control connection between the client and server and sends commands over this in ASCII or ordinary text!



```
Subject Tree
   A_List_of_Contents
»  A_List_of_Subject_Trees
   About the Subject Tree
»  Arts & Humanities
»  Biology subject tree
»  Chemistry
»  Civil-liberties
»  Computers
»  Conferences, (from Norway)
»  Drugs
»  Economics
»  Environment
»  Geography and Oceanography
»  Grants and Funding
»  History
»  International Organisations
»  LPF

Bookmarks
»  Gopher Information Service
»  gopher.netsys.com
```

```
Status: Found 99 matches -- Ready

corton.inria.fr        /pub/docs/internet   tcp-ip-admin.ps.Z
decuac.dec.com                              tcp-ip-intro.ps.Z
faui43.informatik.
ftp.uni-kl.de
ftp.urec.fr
gatekeeper.dec.com
ghost.dsi.unimi.it
gumby.dsd.trw.com
hwarang.postech.ac
kids.kotel.co.kr

Search Term: tcp
Host: decuac.dec.com
Location: /pub/docs/internet
File:
Size:            Mode:            Date:
```

```
Similar to:

Resulting documents:
1000 10.7K (05/27/94) rparson@sp Re: acid rain - ACS report
 549  3.6K (04/04/94) Tom Gray < Re: --FOE Pushes on Acid Rain
 549  3.6K (04/04/94) Tom Gray < Re: --FOE Pushes on Acid Rain
 463  1.5K (04/20/94) rparson@ri Re: Re: Acid Rain
 415  1.3K (07/02/94) Jack Ferma Re: Re: REQUEST: Information about acid rain for project
 415  281 (04/19/94) mcox@nths. Re: Acid Rain
 402  1.2K (05/05/94) Bill Leenh Re: Acid Rain
 390  1.0K (06/20/94) kerrinp@by Re: REQUEST: Information about acid rain for project
 390  908 (04/06/94) fergusos@J Re: Acid Rain: lake cleanup info?
 292  4.8K (05/31/94) babcock@mb Re: Nucleic Acid Structure Analysis Program Release
 292  4.9K (06/05/94) babcock@oc Re: Nucleic Acid Structure Analysis Program Release
 292  4.9K (05/31/94) babcock@mb Re: Nucleic Acid Structure Analysis Software Release

Status: Found 40 documents.
        Searching environment-newsgroups.src.
        Found 40 documents.
```

Fig. 4 'Screendump' with three applications running

When data is going to move, the client and server open a data connection. The data connection can keep going whilst the user issues further commands.

Electronic mail

E-mail is either the saviour of modern society, and trees, or the devil on the icing on the cake of technocracy. Electronic mail, at its simplest, is a replacement for paper letters ('snail' mail) or the facsimile. Sending e-mail (Fig. 5) is easy if you know the address of the person you want to get it to. You type in the message using whatever facility you are familiar with and then submit it to the mail system (put it in the postmaster's bag!). Then a series of automatic systems (message handlers) will sort and carry it to the destination, just like post offices and sorting offices do with paper mail.

The protocol used for electronic mail in the Internet is called the Simple Message Transfer Protocol, or SMTP. The model is one of message handling systems and user agents all talking to each other, both using the same protocol. The user program invokes SMTP to send to a receiver, which may be a mail relay or actual recipient system.

An SMTP mail address looks like this:

J. Crowcroft@cs.ucl.ac.uk.

The general form of such an address is:

User @ Domain

The Domain is as defined in the Domain Name Service (DNS) for the host implementing SMTP. The DNS name is translated to an IP address. The sending system merely opens a TCP connection to the site and then talks the SMTP protocol.

Mail lists

Some mail system managers use info-servers to maintain mail lists. Mail lists are ways of sending a message at one go to groups of people having a common interest or purpose. They resemble, but are completely different in implementation from, bulletin boards, which are discussed below. Mail lists are very useful when used discriminatingly. On the other hand, because it is as easy to send to a list as to an individual, sometimes users propagate junk mail to large groups of people. The most common piece of junk mail is to do with list management (e.g. 'please add me to this list' or 'please remove me from this list', which should be directed to list managers, usually as '*listname*-request'), but other human errors include sending irrelevant or offensive information.

Bulletin boards

Online bulletin boards are analogous to the pinboards we are all used to from offices and schools. There is a fundamental difference between a '*bboard*' and a mail list: mail arrives in an individual's mailbox, and the individual's attention is drawn to it; bulletins arrive on a bboard, and users decide whether or when they want to read that bboard, if at all.

A less fundamental difference is the protocol. Bulletin boards are effectively a single mailbox. Thus the overhead of delivery in terms of computer storage is much lower for a bboard than for a mail list.

Archie

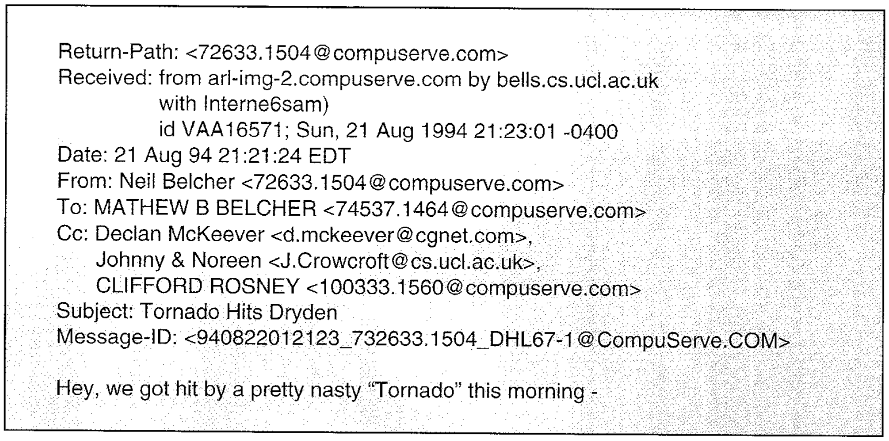
Archive servers appeared in the mid 1980s. Initially, they were a logical extension of FTP servers. They provide indexed repositories of files for retrieval through a simple protocol called Prospero.

Archive servers had been in place manually for some time, simply as well maintained FTP servers. The first attempt at automating co-ordination was to use a simple protocol. This involved periodically exchanging a recursive directory listing of all the files present on a given server with all other known servers. Thereafter, access to a given server for a file present on another could have two results: either the client could be redirected to the right server, or the current server could fetch it and then return it to the client.

These two approaches are called 'referral' and 'chaining' in some communities, or 'iteration' and 'recursion' in others. These ideas are discussed further below.

Whois and Finger

Whois is one of the oldest and simplest information servers in the Internet. It allows you to look up someone's e-mail address, and other information that a user may be



```
Return-Path: <72633.1504@compuserve.com>
Received: from arl-img-2.compuserve.com by bells.cs.ucl.ac.uk
          with Interne6sam)
          id VAA16571; Sun, 21 Aug 1994 21:23:01 -0400
Date: 21 Aug 94 21:21:24 EDT
From: Neil Belcher <72633.1504@compuserve.com>
To: MATHEW B BELCHER <74537.1464@compuserve.com>
Cc: Declan McKeever <d.mckeever@cgnet.com>,
    Johnny & Noreen <J.Crowcroft@cs.ucl.ac.uk>,
    CLIFFORD ROSNEY <100333.1560@compuserve.com>
Subject: Tornado Hits Dryden
Message-ID: <940822012123_732633.1504_DHL67-1@CompuServe.COM>

Hey, we got hit by a pretty nasty "Tornado" this morning -
```

Fig. 5 Example of electronic mail

happy to give away, simply by knowing their name (Fig. 6). Originally, it was a purely central server run on the ARPANET for all managers/contacts of networks attached to the ARPANET for the DCA (Defence Communications Agency) (RFC 954 — see Bibliography).

Basically, a Whois server runs on TCP port 43* and awaits simple command lines (in ASCII text, ending with CRLF†). The server simply looks up the command line or 'name specification' in a file (perhaps using fuzzy or soundex matching) and responds, possibly with multiple matches. Whois is for keeping organisation contact information.

Note that each returned entry has an 'NIC' handle to distinguish it (i.e. to act as a unique key). NIC is the Network Information Center, of which there are now many around the Internet.

```
whois -h ftp.ripe.net bates
person:     Tony Bates
address:    RIPE Network Coordination Centre (NCC)
address:    PRIDE Project
address:    Kruislaan 409
address:    NL-1098 SJ Amsterdam
address:    Netherlands
phone:      +31 20 592 5064
fax-no:     +31 20 592 5090
e-mail:     Tony.Bates@ripe.net
nic-hdl:    TB230
notify:     Tony.Bates@ripe.net
changed:    Tony:Bates@ripe.net 931230
source:     RIPE
```

Fig. 6 Example of Whois output

Finger: The Finger protocol derives from the document RFC742. A Finger server runs on UDP or TCP port 79. It expects either a null string, in which case a list of all people using the system is returned, or, if a string is given, provides information available concerning that person (whether logged in or not).

Some people find Whois and Finger alarmingly insecure. One particular scare in the Internet concerning security was due to a simple, but extremely effective, gaping bug in the most widely used implementation of the Finger server; this may be why it scares some people.‡

DNS

The Domain Name System is really designed as a Network Information Service for internal use by tools rather than directly by users. However, the names it holds appear in location information currently used by many services and are also the basis for electronic mail routing.

The DNS model is that all objects on the net have a name that should be that given by the people responsible for the object. However, this name is only part of the full way to specify the object. The fully distinguished name is part of a hierarchy of names, which are written as per a postal 'address', for example:

swan.computer-lab.cambridge.ac.uk

The 'top level' is a country code (e.g. '.uk') or US Specific (e.g. '.com'). Any organisation owns its own level and the names of the levels below. Any string is usable. Aliases are allowed — names more friendly than addresses.

Any owner of a name space must run a server. The owning site then must inform sites at a 'level' above where their server is. At the same time, they tell their server where the level above is.

Applications (FTP, Mail, TELNET, Mosaic, etc.) use a library function to call the resolver. They give the resolver function a name and it sends a request to the local site DNS server. The DNS server responds with either:

(*a*) the answer
- from its own tables
- from another server it asked on the user's behalf. This is called chaining.

(*b*) a site which can answer. This is called referral.

The Domain Name System holds general purpose 'resource records'.

Wide Area Information Server — WAIS

The Wide Area Information Server idea is based on a search model of information, rather than a browse one. Sites that run WAIS servers have created a collection of *indexed* data that can then be retrieved by searches on these indexes. The access protocol to WAIS servers is based on the standard developed for library searching by ANSI (American National Standards Institute) with the unlikely title Z39.50 (also known as Information Retrieval Service and Protocol Standard).

WAIS has four parts (like most information services except the richer WWW): the client, the server, the database and the protocol.

Client programs (e.g. the X Windows client xwaisq) construct queries and send them using the protocol to the appropriate server. The server responds and includes a 'relevance' measure for the results of the search match to the query.

The actual operation of the protocol is quite complex, as it permits exchanges to be broken into separate parts.

* The TCP port numbers direct data to the appropriate application software program.

† CRLF is 'carriage return' (ASCII character 13) followed by 'line feed' (ASCII character 10).

‡ This bug is long since fixed. Basically, the Finger daemon had storage for receiving a limited request/command, but could actually be handed a larger amount of information from the transport protocol. The extra information would overwrite the stack of the executing Finger server program. An ingenious hacker could exploit this by sending a Finger command carefully constructed with executable code that carried out his desired misdemeanour. The problem was exacerbated on many systems where the Finger server ran as a special privileged process (root!), for no particular reason other than laziness of the designers of the default configuration. Thus the wily hacker gained access to arbitrary rights on the system.

Table 1: Gopher response types

0	Item is a file
1	Item is a directory
2	Item is a CSO (qi) phone-book server
3	Error
4	Item is a BinHexed Macintosh file
5	Item is DOS binary archive of some sort Client must read until the TCP connection closes. Beware
6	Item is a UNIX uuencoded file
7	Item is an Index-Search server
8	Item points to a text-based telnet session
9	Item is a binary file! Client must read until TCP connection closes. Beware
+	Item is a redundant server
T	Item points to a text-based tn3270 session
g	Item is a GIF format graphics file
I	Item is some kind of image file. Client decides how to display

WAIS permits retrieval of bibliographic as well as contents (including images) data.

A search request consists of seed words (or 'keys') typed by the user into the client, together with a list of documents (identified by a unique global ID). The response is quite complex and includes a list of records, including the following fields:

headline — basically a title/description
rank — relative relevance of the document
formats — list of formats available (text/postscript etc.)
document ID
length

Gopher

Gopher is a service that listens for TCP connections on port 70. It responds to trivial string requests from clients with answers preceded by a single character identifying the type, a name and a selector. The client then chooses what to do and how to display any actual data returned (Table 1).

World Wide Web

The World Wide Web makes all these previous services look like stone tablets and smoke signals. In fact the Web is better than that! It can read stone tablets and send smoke signals too!

The World Wide Web service is made up of several components. Client programs (e.g. Mosaic, Lynx) access servers (e.g. HTTP Daemons) using the protocol HTTP (HyperText Transfer Protocol). Servers hold data, written in a language called HTML, the HyperText Markup Language. As indicated by its name, it is a language (in other words it consists of keywords and grammar for using them) for marking-up text that is hyper!*

The pages in the World Wide Web are held in HTML format and delivered from WWW servers to clients in this form, albeit wrapped in MIME (Multipurpose Internet Mail Extensions) and conveyed by HTTP.

A note on stateless servers

Almost all information servers above are described as *stateless*. State is what networking people call *memory*. One of the important design principles in the Internet has always been to minimise the number of places that need to keep track of who is doing what. In the case of stateless information servers this means that they do not keep track of which clients are accessing them. In other words, between one access and the next, the server and protocol are constructed in such a way that they do not care who, why, when or where the next access comes from.

This is *essential* to the reliability of the server and to making such systems work in very-large-scale networks such as the Internet with potentially huge numbers of clients: if the server did depend on a client, then any client failure would leave the server in the lurch, possibly not able to continue, or else serving other clients with reduced resources.

3 Security, performance guarantees and billing

Improvements in CPU, memory and storage performance/price have made many new applications possible which are finding increasing use as a result of reduced connectivity costs. A corresponding increase in network functionality has yet to happen. The Internet is experiencing a number of problems due to the growth in its size and the breadth of the community using it. These include:

- *scale:* The number of systems is exceeding the range of numbers available for addressing systems in the Internet — a problem similar to the one that everyone in the UK is accustomed to from time to time with telephone numbers. The way these numbers are allocated is also leading to problems with the amount of memory in the router boxes that hold together the Internet. Currently, these need to hold the full list of every site in the Internet. A more hierarchical approach (like the phone system or the postal system) will fix this.
- *security:* Security concerns are broad. The range includes privacy of information, authentication of users and systems to each other, access control to systems and resources, prevention or denial of service, and other kinds of misuse, even including covert signalling. Security is really not a question that is relevant when talking about the Internet itself. What needs to be secured are hosts and information. However, the network must provide relevant hooks for security to be implemented.
- *billing:* Currently, the charging model in most of the Internet is a leasing one. Bills are for the speed of

* *Hyper* comes from the Greek prefix meaning above or over, and generally means that some additional functionality is present compared with simple text. In this case, this additional functionality is in two forms: graphics or other media, and links or references to other pieces of (hyper-)text. These links are another component of the WWW called *Uniform Resource Locators.*

access, not the amount of usage. However, many believe that at least during busy periods, or else for priority service, billing will be necessary as a negative feedback mechanism. This will also require security so that the right people can be billed legally.*

- *guarantees:* The Internet has not historically provided guarantees of service. Many providers have done so, but typically by overprovisioning the internal resources of their networks. In the long run, this may prove viable, but at least for the next few years mechanisms will be needed to control guarantees, especially of timeliness of delivery of information. For example, many information providers, such as the share trading and news businesses, value their commodity by time.

Performance parameters

There are three key parameters to worry about in the network and these are important if you intend using a part of the Internet to deliver commercial or dependable WWW services:

(*a*) *Errors:* Transmission technology is *never* perfect. Even glass fibre transmission has occasional errors.

(*b*) *Delay or latency:* A network is not infinitely fast. In fact, now that we are building a global society, the speed-of-light that Einstein was so keen on is becoming a significant factor. Also, busy networks run slower.

(*c*) *Throughput:* Different networks are built for different amounts of traffic. So although some networks may have lower latency they may also have lower throughput. Normally, though, latency and throughput are largely unconnected. Typically, throughput is a feature of how much you pay, whereas latency is a feature of the distance you are communicating over plus the busyness of the net.

Internet service models

The Internet provides a best-effort service. Data can be sent whenever you want: you do not have to know that there is a receiver ready, or that the path exists between you and a potential receiver, or that there are adequate resources along the path for your data. You do not have to give a credit card number or order number so you can be billed. You do not have to check the wire to make sure there are no eavesdroppers.

Some people are uncomfortable with this model. They point out that this makes it hard to carry traffic that needs certain kinds of performance guarantees, or to make communication secure, or to bill people. These three aspects of the Internet are intimately connected and the rest of this paper briefly discusses how research at University College London and elsewhere is leading to a new Internet model which accommodates them.

Best effort and charging: The current model for charging for traffic in the Internet is that sites connect via some 'point of presence' of a provider, and pay a flat fee per month according to the speed of the line they attach with, whether they use it to capacity or not.

For existing applications this model is perfect. Most sites wish to exchange data which has a value that is not increased radically by being delivered immediately. For instance, when I send electronic mail, or transfer a file, the usefulness to me is in the exchange.

The network provider maximises profit by admitting *all* traffic and simply providing a fair share. As the speed decreases, the usefulness to me decreases, so I am prepared to pay less. But the increase in possible income from the additional users outweighs this. The underlying constant cost of adding an additional user to the Internet is so low that this is always true.

However, there are other kinds of traffic that this 'best effort' model does not suit at all and these will now be considered.

Real-time traffic: The Internet has been used for more than 4 years now to carry audio and video traffic around the world. The problem with this traffic is that it requires guarantees. It has a minimum bandwidth below which audio becomes incomprehensible, and even compressed video is just not usable. For human interaction, there is also a maximum delay above which conversation becomes intolerable.

In the experimental parts of the Internet, we have reprogrammed the routers which provide the interconnection to recognise these kinds of traffic and to give it regular service. There are two aspects to this. Firstly the minimum bandwidth guarantee must be met and this is done by looking more frequently at the queues of traffic in the net for traffic that needs more capacity. The delay seen by this traffic is affected by the variations in the other traffic on the network and by the basic transmission time (speed of light, or thereabouts — this is still a significant factor for transmission around the world, however it is one that cannot be altered).

As the other traffic is increased, the video or audio experiences increasing delays and variation of delay. So long as these stay within tolerable limits, the receiver can adapt continuously (e.g. in silences in audio or between video frames) and all is well. The receiver must compensate both for a variation of inter-packet arrival times and for a possibly varying mean delay between sender and receiver. If the mean delay is constant, then this makes things simpler. Meanwhile, any spare capacity carries the old best-effort traffic as before.

However, when the total amount of traffic is higher than capacity, the system starts to fail. At this point there are three views on how to proceed:

(*a*) Engineer the network so that there is enough capacity. This is feasible only while most people's access speed is limited by the 'subscriber loop', or tail circuits that go to their homes/office. When everyone has fibre to the home/desktop, the potential for drowning the

* There are some who believe that every type of Internet access should be billed for on a usage basis. This is problematic, and in fact it has been shown that it does not maximise profit. Only the user knows how 'urgent' a file transfer is. With very many types of data around, the net can only charge for the ones it really knows about, like long-distance voice or high-quality video. Currently, the more users there are, the less share each gets, so that a bill based on the actual connect time is not reasonable and a bill based on amount of data transferred does not reflect the actual value seen to the user.

Jon Crowcroft is a professor of networked systems in the Department of Computer Science, University College London, where he is responsible for a number of European- and US-funded research projects in multimedia communications. He has been working in these areas for over 15 years. He graduated in Physics from Trinity College, Cambridge University in 1979 and gained an MSc degree in Computing in 1981 and a PhD in 1993. He is a member of the ACM, the British Computer Society and the IEE and is a senior member of the IEEE. He is general chair for the ACM Special Interest Group on Communications. He is also on the editorial teams for the *Transactions on Networks* and the *Journal of Internetworking*. With Mark Handley he is the co-author of 'WWW: Beneath the surf' and of 'Open distributed systems', both published by UCL Press.

Address: Department of Computer Science, University College London, Gower Street, London WC1E 6BT, UK.

Internet will be alarming. Note though, that the phone network is currently over-engineered so, for audio capacity, everyone could certainly be switched over to the Internet, and all phones switched over to Internet-based terminals with a flat-fee model. The Internet operates with a flat fee rather than usage-based charging, because it has capacity for all existing applications. If those existing applications incorporate interactive voice traffic, then the charging model still applies, and the advantages of a single network for voice and data is clear.

(*b*) Police the traffic by asking people who have real-time requirements to make a 'call set-up' as they do with the telephone networks. When the net is full, calls are refused unless someone is prepared to pay a premium and incur the wrath of other users by causing them to be cut off!

(*c*) Simply bill people more as the net gets busier. This model is proposed by economists at Harvard and is similar to models of charging for road traffic proposed by the Transport Studies group at UCL. We believe it is optimal. Since we have already reprogrammed the routers to recognise real-time traffic, we have the ability to charge on the basis of logging of this traffic. Note that we can now charge differentially as well. Until the network service offers guarantees which differentiate users' traffic, it is difficult to charge on the basis of usage (for time, or packets, or service). Now that guaranteed services are beginning to be feasible, this sort of charging will become possible. However, in introducing charging in this way, we have maintained all the original advantages of the Internet (no call set-up, easy to rendezvous etc.).

4 Conclusions

In this tutorial we have looked at the service model of the world-wide Internet. We have discussed the kinds of applications that have evolved to make information readily available to non-expert users, and we have looked at the underlying technology used to route and transfer the data around the network. Finally we have looked at some of the areas that are being explored as the Internet is extended to add new services on a more commercial basis.

Bibliography

Details of Internet protocols are given in the RFC (Request for Comments) documents maintained by the Network Information Center at Network Solutions Inc. They are available on the Internet at many sites, e.g. the UK Academic FTP site, ftp.doc.ic.ac.uk.

1 TRANSMISSION CONTROL PROTOCOL, RFC 793, edited by Jon Postel, September 1981

2 INTERNET PROTOCOL, RFC 791, edited by Jon Postel, September 1981

3 STEVENS, W. R.: 'TCP/IP Illustrated, Vol. 1 and Vol. 2' (Addison-Wesley, 1992 and 1994), ISBN 0-201-63354-X

4 PARTRIDGE, C: 'Gigabit networking' (Addison-Wesley, 1993), ISBN 0-201-56333-9

5 HANDLEY, M., and CROWCROFT, J.: 'World Wide Web: beneath the surf' (UCL Press, London, 1995) ISBN 1-85728-435-6