ACSNET − The Australian Alternative to UUCP

Piers Dick-Lauder

Basser Dept of Computer Science
University of Sydney

R. J. Kummerfeld

Basser Dept of Computer Science
University of Sydney

Robert Elz

Dept of Computer Science
University of Melbourne

ABSTRACT

ACSNET is a network with goals to serve a function similar to that currently served by the UUCP network. Routing is implicit, and addressing absolute, with domains. The network daemons attempt to make use of full available bandwidth on whatever communication medium is used for the connection. Messages consist merely of binary information to be transmitted to a handler at the remote site. That handler then treats the message as mail, news, files, or anything else. Intermediate nodes need not consider the type of the message, nor its contents.

1. Introduction

ACSNET is a loosely coupled network of heterogeneous machines, and has a purpose and function similar to that provided by the UUCP network in wide use in most of the UNIX® world.

Before continuing we must make two points. First the matter of naming. Perhaps ACSNET's biggest problem is the lack of a suitable name. The developers (PD-L and RJK) call the software 'The Sydney Unix Network' or SUN for short, while the *network* built using SUN is called ACSNET. Unfortunately, 'SUN' may be confused with a Unix based workstation of the same name and 'ACSNET' suggests some relationship, or at least similarity with the U.S. CSNET network. No other label has gained sufficient approval to catch hold, so ACSNET serves for the time being.

Second, a note on our use of UUCP for comparisons with ACSNET in this paper. The authors have no intention to discredit UUCP, or to belittle its achievement in linking the world's UNIX systems in a manner never before attempted. However, its presence as a current *de-facto* standard for UNIX to UNIX communications places it in a position where we cannot avoid making comparisons in order to illustrate certain points with far greater economy of words than would otherwise be possible.

2. Overview

ACSNET provides a message passing service, from one host to another, possibly utilising intermediate hosts in a store and forward manner. Messages may be mail, files, printjobs, news, or almost anything that can be transferred in a string of bytes.

Routing in ACSNET is implicit, users need only be concerned about the name of the host at which the message is to be delivered. They need not be concerned about which hosts the message might visit on its journey to its final destination. If, for some reason, a message is undeliverable, the network will make every attempt to return the message to its original sender.

Messages can be transported over any medium capable of supporting a connection between two hosts. This may be phone lines, ethernet, X.25, twisted pairs, etc. Preferably, the link should provide a transparent 8 bit data link, but links with only 7 useful data bits can be accommodated (at a slight loss in throughput). The network daemons make good use of full duplex communication channels, transferring messages in both directions simultaneously, providing, in ideal conditions, effective throughputs up to twice that attainable with UUCP.

3. Messages and Handlers

A message is a string of bytes addressed to a handler at one or more hosts. A handler is a process that will receive the messages at the final destination. Typically the handler will

impose some further protocol, often recognising a user name (in some form of representation) that the message is directed to. A message, of itself, is addressed merely to a host and a handler.

The notion of messages addressed to handlers is one of the primary differences between ACSNET and UUCP. UUCP functions as a remote command execution system built upon a file transfer protocol. Mail, news, etc. are transmitted by sending the content of the item to the remote system as a file, then sending a request to execute the remote mail, or news, receiver with the file as standard input. ACSNET simply transfers a message addressed to the mail handler on the remote system.

ACSNET uses trailer protocols, where the *header* follows the message. This allows file copying to be avoided on intermediate hosts when routing statistics are updated. ACSNET only ever performs disk to disk copying of a message when it is being copied to its final destination, and optionally, when queueing the message in the first instance.

Currently handlers exist for mail, news, file transfer, and remote printing. A remote command execution handler could be added if the security issues could be adequately solved. Any other handler could be created just as easily, for any purpose that the sending and receiving hosts agree upon.

Unlike UUCP, it is not necessary for intermediate hosts to know of the new handler for correct functioning.

4. Addressing

All messages carry a destination host address. This is the ASCII name of the destination host. Messages also contain a source host address, and may contain a user address. This last item may be anything that the handler requires for its functioning.

Messages may be addressed to more than one host. A copy of the message will be sent to each host addressed, and that will be done with the minimum possible message traffic (or something approaching it). A message may also be broadcast to all hosts. This is most often used for network management messages, such as new hosts connecting, and similar events.

Users also have the option to guide their message through a specific set of hosts. Primarily this is used for network testing, loopback messages would otherwise be impossible to create.

Such a route is expressed in a notation borrowed from UUCP as

host-1!host-2!host-3 ...

However, note that there is a fundamental difference between this form of addressing and UUCP addressing. Each *host*-N is the absolute address of some site. The **!** does not imply a link between two adjacent hosts. In the above example, the message will visit, in order, host-1, host-2, and host-3. But the route taken to travel from host-2 to host-3 is not specified, and in fact, given a suitable topology, the message might travel that route via host-1!

Strictly, in all the above, the term ‘host name’ should be replaced by ‘domain specification’, but that is harder to type, read, say, and think about. Any of the places where a host name was specified, ACSNET would really expect a domain specification. Domains might simply be host names, or they might specify local sub-domains, or perhaps a domain that is not within the ACSNET network, in which case the message will be sent to an appropriate gateway.

Note, nowhere here has it been specified what syntax should be used by users in communicating with user agent programs. It is to be expected that

user@domain

will be the most common format, though the older Australian net syntax of

user:host

will be supported into the distant future. Almost none of the ACSNET code either knows or cares what syntax users will use to send messages.

5. Routing

Each host maintains two tables† to contain network information. The first of these is the network state file, and contains for each known host, a list of the domains to which it belongs, a list of the hosts to which it is directly linked, and the cost and current status of each of those links. One of the domains is special, and is considered to be the primary domain of the host.

The table can also contain various other information, such as a human understandable description of each host, and statistics on messages and bytes sent, received, and passed through. This information is optional, and probably would be deleted on a small host.

The state table is considered public information, and is sent to any host that requests it. It is broadcast to all hosts whenever a new link is added.

† For ‘table’ read ‘file’.

The second table is the routing table. This table indicates which link should be used to transmit a message bound for any host or domain on the net. It is built from the state table, usually whenever that is altered. As each message arrives on a link it is passed to a routing process. That process passes it to its appropriate handler if this is (one of) its final destination(s), and queues it to be transmitted on the next link if the message has further to go. The routing table is used to make this decision.

Information on which links to transmit a broadcast packet that originated at any particular host is also retained here. This is arranged so that broadcast packets travel over a minimum traversal of the graph, and implements Dalal and Metcalfe's extended reverse path forwarding algorithm.‡

This table also contains miscellaneous information, such as local aliases for hosts on the network. It is private information, and is not exported to other hosts.

Routing messages are broadcast to all hosts in the sender's primary domain whenever a link changes status (goes up or down). These are brief messages, indicating the nature of the change, and carry a timeout age, after which they are deleted wherever found. This works well, as typically, distant parts of the net are not interested in local changes, which often might have become outdated before the message reached the host. Rerouting to avoid links that are down can usually be handled by nodes relatively close to the broken link.

6. Gateways

The routing table for any local link can indicate that a non-standard spooling program should be used to send a message over a designated link, or to deliver a message to a particular domain. This can be used to build interfaces to newer, or older, versions of the network software, or to implement a gateway to a foreign network. The spooling program is responsible for performing any transformations required of the message to meet the standards required by messages entering the new network.

This performs admirably when interfacing to a network with similar capabilities, but is less of a success connecting to UUCP for anything but mail, as there is no standard way of performing possibly multi-hop file or news transfers.

7. Calls and Daemons

ACSNET uses node to node daemons to transfer messages from one host to another. Unlike the UUCP uucico process, ACSNET daemons have no knowledge of how, or when,

‡ Y. K. Dalal and R. M. Metcalfe, CACM, Dec. 1978.

to connect to a remote host, except in the trivial case where that is accomplished by opening a tty compatible special file. To handle other cases, the routing table may contain the name of a process to run to establish a connection to another host. That process is expected to make the connection, then exec the daemon with standard output open (read-write) to the remote host. This permits easy expansion to a wide variety of possible connection types.

A pair of daemons transfer data between themselves over three channels in each direction simultaneously. This allows up to 6 messages to be in flight between any pair of nodes at any one instant. Messages are assigned to one of the three channels based upon their size. This allows small messages to overtake larger ones on another channel, and prevents those extraordinary delays that can occur when a particularly huge message is being transferred.

The daemons also keep track of the current position in each message in transit. This allows messages to be restarted without transmitting data that had been received correctly at its destination, should a link die prematurely, or a system crash.

Messages on each channel are sent via a windowed packet scheme, similar to that used by HDLC (and UUCP, and X.25), and are checksummed using the standard CCITT CRC-16 algorithm. This checksumming can be disabled for a link if it is known to be reliable, typically if ACSNET is used over another protocol. Messages can also contain end to end checksums, to guard against corruption while waiting at an intermediate node.

8. Status

ACSNET is currently being used on VAX 11/780 and 11/750 processors, PDP-11/34's, Sun Workstations, Perkin-Elmer, Plexus (P60), ELXSI 6400, Gould, IBM PC, and other less widely known machines. These processors are variously operating under V7, 4.*BSD, Systems III and V, and Venix/86. A VMS version has been suggested, but at this stage, not attempted.

There are about 200 hosts on the Australian network, and though a few of these currently use the previous software, the remainder will probably convert in the near future.

9. Todo

There will be a message disassembly, reassembly facility, to permit huge messages to be transmitted without overloading intermediate nodes.

A UUCP gateway is needed. This is hard because of problems with multi-hop file transfers.

Some tuning remains to be done. One possible improvement on System III and V systems, and 4.2BSD, would be to use the available interprocess communication mechanisms (named pipes, sockets) to allow the routing process and handlers to become daemons, and be created just once, rather than once per message.

10. Availability

The source code is available under license. Anyone interested should apply to Bob Kummerfeld at the address above.

11. Conclusions

ACSNET is a suitable network system for connecting comparatively large networks, of a size comparable to the UUCP network, that may operate in a relatively unmanaged environment. Its implicit routing makes it considerably easier to use than standard UUCP, and more accurate and adaptable than the heuristic UUCP routing algorithms now becoming available.

We feel that ACSNET would be suitable as a general replacement for UUCP.