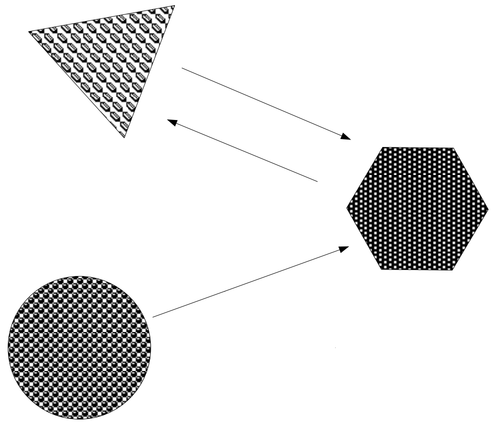
Interconnection of group communication services

Harri Salminen

hks@funet.fi



What is a group communication service?

Any service designed to support communication between a group of persons. In this paper I will concentrate on the interconnection methods between some popular systems that are being used in FUNET to offer group communication services, namely various flavors of mailing lists, netnews and PortaCOM. I will describe how to interconnect them although the methods presented could probably be applied to most systems[6] that support some form of the RFC 822[2] or X.400 [1]interconnectivity.

Mailing lists

Mailing lists were created soon after electronic mail was invented to distribute messages to a list of people that was predefined with a short name so that the originator didn't have to type in all the addressees individually. Simplest of these mailing lists are local to a user or a machine so that no-one can't send to them via a network. Obviously this was unsatisfactory solution for unmoderated communication in networked communities and the solution was to create pseudo users or aliases which redistribute everything sent to them. There's two major approaches to mailing lists in academic networks that have different historical backgrounds:

The Internet mailing list approach

This is the convention that was created among the users of DoD ARPA Internet. In it the mail message is distributed using the RFC821[1][5]. protocol and addressing without manipulating the RFC 822 To: or Cc: fields. Normally only a Sender field should be added to point to a **human** responsible for taking care of the problems the automatic distribution might create. In reality there might sometimes also be Return-Path, Resent-From, Errors-To, Autoforwarded etc. headers to identify that it was automatically forwarded but a much more common case is that there's no added headers at all and the Originator will get the error messages instead of the list maintainer. By convention this list maintainer can usually be found behind a *listname*—request alias in the same host as the list itself but it's not necessary for the list to function, although it helps others to

[1] Also called SMTP or more generally MTA-MTA protocol in this paper

find him. The benefit of having the list's address in the To: or Cc: field is that you can in most mail user agents do a reply to all addressees of the message including the list or just reply to the originator of the message. Also you can find out from which list the message came from. Only some really broken gateways might resend this message back to the list creating a loop. Currently almost all mailers deliver mail based on the MTA-MTA level addresses and only in case of problems they send error messages back to the sender or in some cases to the originator. Since error messages are not at all standardised in RFC 822 they vary widely and often fail to even mention which kind of address or even which kind of letter caused the problem...

Internet also supports moderated lists where a human moderater receives all the messages and after editing resends them using a non-public list to the subscribers. Often many messages are packaged together as digest in a standard form[7] that can be later split back to separate messages in a User Interface or even in a gateway. Because of the moderation work needed this method isn't as common as normal open lists and I'd recommend digesting only if you are going to be adding real value by editing like in a magazine and not just repackaging together everything unedited. These can be gatewayed to netnews and PortaCOM in whole or in pieces and contributions to the editor can be directed back to the moderator. Often there's no need for any gateway since the moderator might send the digest directly to the different forums.

This Internet approach to mailing lists has spread widely so that almost every RFC-822 mailer can support them and even many X.400 MTAs support it. So Internet style mailing lists are in use almost everywhere. Archives are normally maintained by concatenating messages to local files and putting them available on anonymous ftp or in some cases via separate mail archive server. Very large lists can cause high loads to the distributing host and networks which has been eased in some cases by setting up local sublists. More efficient network use could also be done by source routing a group of addresses to a some agreed mailer nearer the destination but determining which mailer is nearer to the destination might be a complex task. Despite of all the shortcomings this is the easiest solution for small static groups and works almost everywhere.

The Listserv approach

The other major approach to mailing list distribution has it's roots in BITNET and EARN. First before Internet addressing, mailers or even RFC-822 headers required for mail gateways to other networks had become commonplace, a virtual machine called LISTSERV was created in BITNIC[2] to redistribute mail and IBM NOTE *files* using NJE addressing that uses eight character usernames and nodenames. Among others places this original LISTSERV was also installed in a node FINHUTC where I tried to extend it's capabilities for local use. Another person who had noticed the shortcomings in the original LISTSERV was a bright young systems programmer *Eric Thomas* who worked with the FRECP11 system in Paris. He decided to write a better one from scratch in his spare time which was then called *the Revised Listserv*. For short it can be called just the Listserv here since the original one isn't used even in BITNIC anymore.

During the years there has been much development but there's still many systems on EARN and BITNET that don't have mailers or use IBM NOTEs. Since one can't expect that users always send mail via mailers a LISTSERV node has to have dummy users for each list they support. This rules out in most cases the use of long addresses with —request suffix. Eric Thomas sees the Listserv as a mail based server application and not as a part of a mailer. So listserv processes the files it receives on the dummy userids more than a mailer proably would. Most importantly the default action is to remove most of the original headers except the Subject and From, add a To header pointing to the subscriber and a Sender, in most cases also Reply-To, pointing to the list. This results in clean short headers for the user which can easily be archived based on the Sender: field in a popular RFC 822 user agent under the VM/SP CMS operating system. Unfortunately these short headers also can cause problems which might be hard to solve. Especially since the RFC-822 error messages are not at all standardised, except that they should be send to the Sender: address, they have often caused mailing loops. Eric has mostly solved this problem by developing very extensive loop detection algorithms that catch most error messages and even duplicated files and forwards them to the list owner so that nowadays real mailing loops are fairly uncommon.

Listserv has gained enourmous popularity in VM/SP CMS sites all over EARN, BITNET and other NJE based networks because of it's extensive range of features and automatic functions ranging from automatic subscription by the user to a complex SQL style database supporting mail and file archive functions. Listserv has now many functions and features that can be used to have quite

[2] BITNET Network Information Center

Internet style headers, lists closed for even submission outside it's members, personal mail forwarding, providing network information services, line monitoring, file traffic control etc.

First approach to network overload was sublists like in the Internet but soon a peering technique was developed in which lists are linked bidirectionally to each other in equal fashion. Since administering lots of peer links is timeconsuming and error-prone another more automatic distribution optimisation technique was developed. Now most of the major LISTSERVers belong to a so called DIST2 backbone where mailing list is automatically expanded in the nearest listserv to the user.

Pros:

- Very good selectivity and reachability
- Closed an moderated list can be supported
- Especially suited for small static groups
- One familiar mail user interface is enough for all electronic communication

Cons:

- High volumes and missing structure can cause information overload to the user
- User has to know how and where to subscribe and contribute
- No way to cancel or expire messages
- Loops, expired accounts and other errors cause many problems especially for the maintainers
- Distribution might cause unnecessary load to the network

Netnews

Due to the many problems with large mailing lists in a relatively slow and unreliable uucp network a standardised message distribution system with hierachy and automatic loop control was developed. Actually loops were even desired to ensure that a message was distributed at least via some route in the event of failures. The messages consist of a relatively well defined RFC-822 subset and standardised control messages. The RFC 1036[3] defines the following set of required headers: From, Date, Newsgroups, Subject, Message-ID and Path. There's also defined a set of optional headers: Followup-To, Expires, Reply-To, Sender, References, Control, Distribution, Keywords, Summary, Approved, Lines, Xref and Organisation. Others can be used as well and like in RFC-822 they should be passed through unchanged. Especially important for the interconnection are the Newsgroups header which explicitly categorises the message to a predefined group, the Message-ID which is used in association with a history database for automatic loop control and duplicate removal, and the Path which tells which route the message has traversed.

Because of it's origin there's a wide variety of News Transfer and User Agents for UNIX systems but also increasing number implementations for other operating systems like VM/SP and VMS.

Pros:

- Optimized for large public distribution
- Easy and private group selection and browsing
- Support for group hierarchies, keywords and references
- Message cancellation and expiry control is possible
- No need to administer individual users
- Distributed control without a single central authority possible
- Coordinated, redundant and optimised loop free distribution network

Cons:

- Not available for everyone without a gateway
- Closed membership groups not normally supported
- Not suitable for small widely distributed groups
- High volume leads to short expiration times

PortaCOM

PortaCOM is a portable version of COM which is a computer conferencing system developed originally at QZ. It has a centralised database where all messages are stored along with lots of pointers between them for referencing. Messages are grouped to Conferences that don't have any hierarchy but long descriptive names. In addition all comments are rigidly linked to other messages to form a comment tree. It has it's own intergrated user interface and is normally used by remote login to a single central computer from all over the network. The PortaCOM NICE interface offers a possibility for external mail links between different PortaCOM conferences or to remote mail users. PortaCOM converts the name of the originator from the internal form to a valid Internet address for the From field. To field points to the conference and the recipient in a modified Internet mailing list style and Subject has of course the same meaning as elsewhere. Message-ID is also used for a reference identifier and loop control like in netnews. In-Reply-To contains a reference to the Message-ID of the commented message and X-Envelope-To contains the MTA level address. Archival is sometimes done after the database fills up by setting up a separate archive COM or extracting messages to files.

Pros:

- Real time operation except for external links
- All comments are linked together to form trees
- Closed and Public groups are easy to create and maintain
- Messages can easily be canceled later
- Central administration and backup procedures on one system

Cons:

- User often has to leave his home environment
- Only few centralised systems are in use without much interconnection
- Not many choices for a user interface
- Costs real money to license
- Not easy to modify or extend locally

How they can be interconnected?

For interconnecting these three quite different approaches to group communication we have to look for the required and optional common attributes that could be mapped to each other. Also loop control and good error handling are necessary features for a reliable interconnection. The good old rule of thumb for well working mail systems "Be liberal in what you accept and strict in what you output" is even more important in group communication interconnection since the message has already been distributed to one community and shouldn't anymore just be returned to the originator for corrections or in the worst case just logged in some log file as "bus error, core dumped" and lost leading to partitioned discussions and loss of information to the users. The following represents mainly my view on how the attributes present in different group communication systems should be handled.

Matching attributes

Conferences, *newsgroups* or *mailing list addresses* define specific fairly static group communication activities or discussion forums. These are the entities that can be interconnected together with a gateway. The naming conventions vary greatly and they have to be mapped case by case to each other by the gateway. Fortunately they are fairly static and the user needs to know only the one used in his favourite system. To:, Cc:, Resent-To:, Resent-Cc:, newsgroup etc. headers normally contain only information used as destination inside one system so they should normally not passed through although additional recipients might sometimes be of some informational value.

If a message is *crossposted* to a several different forums at the same time there's a possibility that the copy arriving later to the gateway will be discarded by the loop control systems as a duplicate. In case only one of the forums was linked the replies most probably will not arrive to the unconnected forum. This hard to solve and would need a forum-ID embedded in the message-ID or separate history databases for each forum along with full linkage of all related forums. For now on one has to accept that his message might not be distributed on all parallelly gatewayed forums unless sent separately to each of them. The reply, followup or comment might not also reach those on parallel forums like the original at least partially might have done.

Originator of the message is a natural requirement at least for information. It can be fairly easily be fullfilled because all systems support RFC—822 style *From* addresses although sometimes the domain part isn't a valid internet domain but some unofficial one. A good gateway can try to make the addresses easier to reply by looking at a mapping database. For unknown non—internet addresses the gateway can try to help by constructing an indirect route via some known host in found in the path. Personal name is even more informational in the nature and should be mapped to the netnews subset of RFC 822 for compability. PortaCOM messages don't have a separate personal name since it's already nicely available as the userid in the address. Resent-From addresses are used only in mail systems but they are allowed as extra informational headers inside other systems.

Since electronic addresses can often be very cryptic almost all netnews articles contain the optional header *Organisation* that should contain the originator's organisation. News systems normally supply a default one if the user hasn't specified it. Since it can be quite helpfull information it should be passed though the gateway and allowed also in the incoming mailings. If the incoming message doesn't contain one,the gateway has the option to supply a default one based on which mailing list or conference the message is coming from.

Date is also very fundamental for our fast moving society and is usually also ready in RFC-822 compatible format. Sometimes it has to be converted to a more cleaner format which can be problematic if it has unrecoverable errors. Also there's often timezone abbreviations that are not widely known so it's advisable to convert all zones to either GMT or +0300 style format. A strict interpretation of RFC-822 would allow only for US timezones, GMT and the one character US military timezones for the rest of world. Especially Bnews 2.11 has serious flaws in it's time zone code and will in addition to improper conversions reject messages with timezones it considers invalid. I think in the worst case the gateway should add a new date to replace the unknown one. Time must go on. Even if it's a bit late . . .

Subject: This has the same informational meaning in almost every electronic communication system, except that it can be missing on some mail systems (most notably IBM NOTE) or otherwise left empty. Since the Subject is a required header at least in netnews a gateway should in that case insert a dummy Subject (none) or maybe the first few words of the body and ellipsis . . . If the message is a reply to an earlier one the Subject should be prefixed with Re: and In-reply-To: or References: field used to reference the message being replied to.[3]

Message-ID is a unique identifier of a message in the network that is used for referencing and loop control. RFC-822 defines it to be of the form <unique@full_domain_name>. Characters allowed in RFC-822 atoms should be O.K. but since some systems create unconformant message-IDs or don't accept all conformant ones the gateway should be prepared to "fix" the most common ones and send error messages via mail about the rest. Although message-ID is not a required item in RFC-822 it's generated by most mailers and should be added at the gateway host if it isn't present. Most notable exceptions are the Crosswell Mailer and LISTSERV which normally even removes message-ID headers. For gatewaying purposes it's enough to set a FULLHDR option for the subscription, after which LISTSERV does minimum "cleaning" and even generates a message-ID if it's missing. If a bi-directional link to a system that removes or destroys message-IDs is made the gateway can't regognize duplicate mailings arriving to it.

In-Reply-To: and *References:* field are used for making backward references to earlier messages. In-Reply-To identifies the message for which the reply was made and normally contain's some free form descriptive test and the message-ID. PortaCOM excepts it to contain only the Message-ID of the earlier message this is a comment for so the gateway has to clean the field to get the PortaCOM comment trees formed. Of course if the message-ID isn't available it can't be referenced and automatic reference search isn't possible. References to other earlier messages should go to the references field in similar fashion. Netnews has slightly different interpretation and doesn't use the In-Reply-To field at all, instead it appends the message-ID of the followed up message to the References: field. Ideally a gateway should convert back and forth between these different conventions but currently it just checks for the correctness and length of the fields and merges them to a common Refences line. I plan to correct that in a future version though.

Sender should have an address of a **human** responsible for the message distribution. According to RFC-822 this is the primary address for sending error messages and problem reports. For a standard LISTSERV list this a problem because the Sender points to the list itself and if the extensive loop catching

[3] The none is used in the current gateway code but I used the latter in the BITNIC listserv enhancements I made and also mh uses it to fill in short subject lines so maybe I'll implement it . . .

mechanism doesn't regognize it in time it will result in a loop. Unfortunately it will sometimes catch too well preventing distribution of large digests or other messages that contain suspicious looking lines in their bodies. The cure for this is to configure the list with Sender pointing to a human and relaxing the loop checking rules by using the optional Sender and LoopCheck keywords with suitable parameters. Normally a new sender field will be generated for the outgoing mail by the gateway so that possible errors with the interconnections will reach the right persons.

Trace information is usually recorded along the route of the message and is mostly usefull for resolving problems. Netnews records only the *path* message has traversed by prepending host names before a userid in the traditional uucp style. Sometimes it's also used for loop detection so the hostnames should be unique registered ones. Although it should not be used for replies it can be used as a basis for forming a possibly working From address especially in case the original isn't a widely recognized one.

The message *body* according to RFC-822 is an uninterpreted arbitrarily long text consisting of any seven bit ASCII character combinations except CRLF. In practice there are arbitrary limits to the length of the message which can cause real problems by truncating messages. A recommended limit for splitting messages is 50–60KB but some systems will choke on much smaller messages. Special care should be taken if a portaCOM conference is linked since the system might have been configured to accept only fairly small messages. Also some systems might not like some control characters like null or some other character combinations. If some system uses eight bit character codes the most significant bit will probably be stripped along the way making the message at least harder to read. Of course other character conversions needed for different systems can cause problems as well but that's a fact of life in networking.

Unmatching attributes

Rest of the attributes don't map well from one system to another so they should either be removed as being unnecessary and out of context or passed through for their possible informational value. It's often matter of taste what is usefull information and what is not. I support the view common in netnews environment that the user should be able to decide if something is important by offering him at least enough information. Many good user agents support user defined filtering patterns but they can't reconstruct deleted information. In the following are some examples of headers and what could be done with them.

Resent-From, Resent-Date, Resent-Subject, X-Resent-From ... are headers that should at least be passed through from *mail* because otherwise reader's wouldn't know who forwarded the message to the list or conference and possibly added some comments.

Control, Expires, Lines, followup-To and Distribution are used in *netnews* would be out of context outside so they can normally be removed. Control messages that aren't normally shown even to netnews readers shouldn't be passed though either since they can't be used to control the outside world anyway. Especially message cancellation doesn't work across a gateway.

PortaCOM is quite sparse with headers and there's normally nothing to remove or add if the *X-envelope-To* header and To headers were processed earlier. Portacom can support all other types of headers by putting them in the beginning of the message body prefixed with % sign E.g. %Organisation: “My real organisation”.

Other headers like *Approved, Summary, Keywords or other possibly user defined ones* can be passed through unchanged, removed or otherwise processed depending on the local configuration and the gateway maintainer's views.

Links and loop control

To identify the incoming mail to a gateway belonging to a certain forum it's safest not to try using some strings in Sender, Newsgroups, To, Cc, Resent-To, Resent-Cc etc. for selecting the right mapping. It's much better to set up a separate address for each different link using the alias system so that both ends of a bi-directional link have unique addresses. For example the following could be in an alias file for a newsgroup dist.main.sub that has links both to a mailing list and a PortaCOM conference:

```
lst-dist-main-sub: "!/usr/lib/news/gwbin/mail2news
   -n distribution.main.sub -d distribution
   -x listgw"
com-dist-main-sub: "!/usr/lib/news/gwbin/mail2news
   -n distribution.main.sub -d distribution
   -o 'PortaCOM Organisation' -x comgw"
```


This way even a message sent by Bcc: to an Internet style mailing list would be mapped to the right newsgroup and also redistributed out of the other link to the PortaCOM system without any regular expression matching or AI techniques. The —x flag is important for the gateway because it excludes the outgoing link back to the mailing list from the distribution. This functionality isn't, to my knowledge, currently available in LISTSERV or PortaCOM which means that they will send the message back to the gateway which will silently discard it using the Message-ID based loop control in the netnews system. Even if the Message-ID is lost or changed the message will be duplicated only once and not left looping around. Untill this kind of exclusion capability by incoming address is possible in the other end of a connection the returning messages are unavoidable although harmless extra traffic.

Each of outgoing link could be defined using the pseudo sites listgw and comgw in the following fashion:

```
listgw:dist.main.sub,!dist.main.sub.all/all:
:/usr/lib/news/gwbin/news2mail listaddress
list-real-address sender contact
comgw:dist.main.sub,!dist.main.sub.all/all:
:/usr/lib/news/gwbin/news2mail conference
 conference-real-address sender contact
```

These will direct the netnews system to select right message and pipe it to the outgoing gateway program, set up link dependent headers like To, Sender, and Received according to the given parameters and deliver the message using MTA level addressing to where it should go. Having real destination and RFC-822 To: header separated is usefull for setting up local Internet mailing lists that actually first come inside news and only after to a separate distribution alias. To the users it will look like the gateway had really sent it to the list.

Implementations

Currently there are many different Unix implementations of gateways between mail and netnews available. Also at least one implementation is available for VM/SP. All the available implementations differ slightly in header processing and interconnection methodology from what I've described here. Many use less complex header mappings, remove most of the headers and don't try to "fix" Date, From or Message-ID values although they are crucial for reliable service. Some try to decipher from the mail headers to which group it belongs and others work well only with local mailing lists or aren't even bi-directional. Still they all have solved somebody's group communication problem.

In FUNET I've been using for production a gateway that was originally developed for the ucbvax by *Erik Fair* <fair@apple.com> with slight modifications. It was further developed and partly rewritten using C by *Rich Satz* <rsatz@bbn.com> who also wrote the gag gateway alias generator for easier configuration of the many parameters in alias and sys files. I've been using that version as a basis for a new gateway that would include the features I've been missing in others. The bbn version is available via anonymous ftp from bbn.com and you might find some gateway versions in nic.funet.fi too. The gateway was originally designed with B2.11 news and Sendmail in mind but it seems to work very well with Cnews and Zmailer too.

Coordination

Because there's no Message-ID based loop control in mailing list distribution systems there should be only one link from netnews to the list. If a list is gatewayed to several different groups in different places, as is sometimes the case today, the message will appear only on that newsgroup it appeared first if the message has got a unique Message-ID already at the distribution point like it should have. By adding your own code to uniquely indentify the gateway in the Message-ID these problems might be circumvented but it might easily break the loop control or referencing.

Uncoordinated gatewaying can cause all kinds of problems so it's better to coordinate with other gateway managers before there's clashes. When you know that you are the only one doing local or national gatewaying of these forums there should be no problems but keep your eyes open since others might start gatewaying the same ones later. After a complete reorganisation of the sfnet, which is a common distribution for FUNET and FUUG, I documented all groups in a single file with a chapter describing the purpose and organisation of each group. The checkgroups messages can automatically be derived from it as well. All in all, group communication services can be interconnected, but you have to be carefull out there.

Future?

I expect the number of links between different systems increase steadily since it's almost impossible to get all users to accept one system for group communication. There's many more different types of group communication systems that haven't yet been interconnected especially outside academic networks but will be sooner or later. If a system can generate and receive some kind of a RFC-822 or X.400 mail the current gateway implementation might need only minor modifications if at all.

One direction of development might be to add more LISTSERV like functionalities for automatical management of mailing lists and optimize their distribution in the Internet and X.400 environments. Currently there's no pressing need for this since the current LISTSERV backbone works quite well although it isn't portable to other operating systems. Extensive new developments might be needed for more advanced group communication support system like those that have been envisaged for X.400 by Eunet, AMIGO[4] or CCITT. These kind of gateways Open the current group communication Systems for Interconnection.

References

[1] CCITT. *Data communication networks, message handling systems, Recommendations X.400-X.430*, 1984.

[2] David H. Crocker. *RFC822: Standard for the format of ARPA Internet text messages*. SRI NIC, 1982.

[3] M. Horton and R. Adams. *RFC1036: Standard for Interchange of USENET Messages*. SRI NIC, 1987.

[4] Uta Pankoke-Babatz, editor. *Computer based group communication: the AMIGO activity model*. Ellis Horwoord, 1989.

[5] Jonathan B. Postel. *RFC821: Simple Mail Transfer Protocol*. SRI NIC, 1982.

[6] John S. Quarterman. *The Matrix: Computer Networks and Conferencing Systems Worldwide*. Digital Press, 1990.

[7] RFC934: Marshall T. Rose and Einar A. Stefferud. *Proposed Standard for Message Encapsulation*. SRI NIC, 1985.

