

# Understanding Mailing List Servers

*Mailing lists were used traditionally for Internet-based discussion groups on special-interest topics. Now corporates are harnessing the power of the medium to disseminate marketing information or technical support to customers, providing benefits for both customers and the company.*

*By Andrew Ward*

**M**ailing lists are widely used on the Internet for both public and private discussion groups on a variety of special-interest technical and social topics, but they also have huge potential in the corporate market, when dealing both with internal and external user groups.

Possible external uses range from technical support to providing customers with sales and marketing information on products and services. Technical support is a good illustration of how list servers can be used for either one- or two-way communication, which can be moderated or unmoderated.

For example, news of bugs and patches could be sent out without any subscriber participation being allowed, whereas a peer technical support discussion group could be used by users to both ask and answer technical questions, possibly overseen by someone from the company who can feed comments back to developers and offer assistance where appropriate. In some cases, such a peer support list would need to be moderated, to avoid unnecessary traffic relating to irrelevant or unnecessary questions.

Of course, to operate a one-way mailing list - a newsletter list - you don't really need a list server. The simplest mail client will generally allow mail to be sent to a group of people, and mail servers will offer even more sophisticated mailing list features and probably also some degree of programmability or scripting. But there are a number of reasons why a company might want to consider using a

dedicated list server rather than the facilities already built into corporate mail systems such as Microsoft Exchange.

Most obviously, where the user population addressed by the mailing list would include large numbers of people who wouldn't already have accounts on the mail server - and the most obvious example of that is present or potential customers within the general Internet population - there would be a significant administrative burden in adding accounts for those people.

## ***In-house***

But even where the users are all sitting on the LAN, there are still good reasons for using a mailing list, and they are generally to do with usability and ease of administration. For example, NTList version 4 can interrogate an on-line database using ODBC and modify the audience or even the content of the messages accordingly. Thus, assuming that a suitable asset database was in place, a support team could create a mailing that would automatically target all users that have a particular make of PC on their desks to inform them of a required software or hardware upgrade.

Even where a mailing list is purely used by the company's staff, they wouldn't all necessarily have accounts on the in-house mail server. Increasingly, companies have to accommodate users who work from home, from branch offices or while on the road, and not all organisations will have the remote access infrastructure to allow

absolutely everyone to have remote access to the corporate network. People will be using a variety of different Internet addresses when working out of the office.

Furthermore, internal systems and intranets are being opened up to business partners, customers and suppliers, and it doesn't seem unreasonable to suppose that a similar progression will happen in the case of mailing list deployment, so what may today be perceived as an internal user population may extend tomorrow to include Internet-based users.

Other technologies such as Web-based discussion forums and conferencing systems can form a better and more productive environment for certain sorts of collaborative working, but mailing lists can still be extremely effective in many circumstances. One of the biggest benefits is that users don't have to run a different program or learn a new interface - they interact with the mailing list via their familiar email client software.

However, mailing list servers have in general suffered from a major problem, which has been a severe lack of ease of use - whether for subscribers, list administrators or server administrators. List server software often dates back to the early Unix-based days of the Internet, when the more obscure the user interface, the happier people were.

List servers also frequently have other problems - for example, with many of them, you need to know what email address you used when you subscribed to the list in order to unsub-

scribe. Since people today have multiple addresses and change addresses from time to time, and mail can be automatically forwarded a number of ways, it's sometimes quite impossible to know which address was used - and it often doesn't appear anywhere in the mail headers.

Fortunately, newer software from companies such as Lyris and Gordano is rapidly addressing these problems. Generally speaking, the interface for list and server administrators is now Web-based, rather than via email. Apart from being a lot faster, the Web interface is much more intuitive and doesn't require use of a huge number of arcane commands, all of which have to be spelt correctly to work.

### **Discussion Groups**

One of the biggest challenges facing businesses today is customer loyalty. If you are offering mortgages or credit cards over the Internet, then your customers - that you could once have counted on keeping for life - are now just a few clicks away from your nearest competitor. One of the techniques employed to encourage customers to stay loyal is to make Web sites sticky - that is, to encourage people to stay there, and to return there.

Increasing the stickiness of a Web site can be achieved in a number of ways such as adding other attractions and offers, but fostering a sense of community among customers is felt to be particularly valuable in some markets. Techniques to achieve this can vary from adding chat facilities to the Web site to providing discussion groups - via a list server - that your customers can join. By swapping ideas and comments about your products, customers reinforce their purchasing decision.

Providing peer support can add value to products sold, as well as help achieve this stickiness. By knowing that they can turn to other users of your product for help, customers are much more likely to be encouraged to buy - and your support burden is reduced. You could even encourage comment on the Web site itself, although this could trigger fast and furious discussion! To return some sense

of control and solicit more structured feedback, a questionnaire can be posted to the list featuring specific concerns that have been raised.

Internally, discussion groups can be invaluable, not just between users but also between support personnel. Part of a network administrator's joining pack could include membership of a discussion group, or a number of discussion groups on different subjects, whose membership would include other people working within support. When you encounter a particularly tricky problem, you can then quickly enlist the help of a group of qualified colleagues.

With an increasing cultural trend within business towards openness and cross-departmental co-operation, a discussion list could be used to allow every department to post regular news, reports or performance information, for viewing by others.

### **Newsletters**

The most obvious use for one-way mailings or newsletters is for advertising, but this is not a good idea - and in some countries quite possibly illegal - unless people have either subscribed to the list explicitly or failed to indicate during some membership or registration process that they don't wish to receive further communication.

Another use for newsletters is to improve the stickiness of a Web site by informing interested parties of any major changes or improvements. This combination of email with a Web site has proved to be particularly successful in certain cases such as FT.com (the *Financial Times* in London).

Internally, newsletters have many

more uses, for example by a human resources department to notify people of a forthcoming change in terms of employment, perhaps to include Internet usage guidelines. It is in the area of human resources that the ability to interface with a database is particularly useful, so that information on pensions and other aspects of employment can be easily targeted to the correct group.

In the area of product support, newsletters can offer considerable value. Registered users can be informed of patches, service packs and new versions, and also upgrade offers. The cost of sending out a mailshot advertising a new version is very little, and the news will often be of value to recipients, encouraging a good response.

Product security alerts and virus news needs to reach users quickly, and this applies equally to internal and external users. Obviously, you will need to inform external users of problems with your own products, but internal users will appreciate being warned of external security threats that they need to be on the look out for, or problems with software that they are using. Users can't be expected to subscribe to virus newsletters or manufacturers' security bulletins themselves.

Mailshots are usually themselves advertisements, but it can sometimes also be useful to include relevant advertisements within mailings sent to discussion groups. NTLlist includes this capability, and can count the number of impressions to form a basis for charging. Advertisements can be rotated, so users don't see the same one on every single message posted to the list.

---

***“A mailing list is not a useful collaborative working tool if all that happens on the users' desktops is that mailing list traffic appears in the inbox along with all other mail.”***

---

# Mailing Lists

## Usability

When choosing a list server, usability is probably the most important attribute, although for some applications - particularly where the list server is used for sending out mailshots - it won't be so much of an issue.

Software is becoming easier and easier to use, and users don't expect to have to spend time learning obscure commands or a difficult interface. Any user interface that is difficult to operate will only lead to an increased support burden.

Subscribing to the mailing list should be easy for the user to undertake. Traditionally, the user has had to send a message containing either the word "subscribe" or the phrase "subscribe listname", via email, to the mail server. This does rely on the user's spelling and correct choice of word, but you can sometimes set up aliases to include alternatives such as sub and join.

More modern servers such as Lyris just require the user to send a blank email to a specific address of the form subscribe-listname@domain, which is much easier. You can make it easier still by including the necessary code in a Web page - whether on the intranet or a public Web server - so all the user has to do is type in their email address and click on a Subscribe button. Some list servers include this capability as standard, and for others you'll need to write a short Web script.

There should be some means of preventing someone from being joined by someone else. This can be achieved by sending a confirmation message to the user to which they must reply, and this process is handled automatically by the list server - all major products have this feature. A closed mailing list that

is restricted to a certain group of users will generally not allow people to subscribe at all. A moderated list can require that members be approved before they are joined to the list.

When accepting subscriptions to the list users may wish to state preferences in terms of how they receive the mailing list postings - whether as plain text or html. Check that the list server not only has the ability to send messages in different formats, but also that users can easily state these and any other relevant preferences.

## Unsubscribing

Far more important than subscribing from the user's point of view is unsubscribing. Traditionally, this has been accomplished by sending an email message containing the word "unsubscribe" to a list server command address, but this is prone to failure. It relies on the user being aware that there is a command address that is separate to the mailing list address, and remembering to use it. It also assumes that the From address in the user's message header is going to be the same as the address they used to subscribe from, which is often not the case.

Implementing a list server that still uses a traditional unsubscribe interface is likely to result in persistent calls to the helpdesk from frustrated users. A Web form that requires the user to specify their email address is also not the answer, since the user may well not remember the address used when first subscribing to the list.

The modern answer is to include within the footer of each message sent to the list a specific address of the form unsubscribe-listname-number@domain, where the number is unique to the user, or a unique URL that they can

click on that will take them to an unsubscribe confirmation Web page. Of course, this means that the message sent out to each user must be unique, and is the technique used by the Lyris list server, for example.

With NTLlist, you can determine the correct name of the list and the email address that the list knows you as by inspecting the message header. With some other products it is possible for a user to obtain a list of all group members and identify themselves, but this is generally disallowed by administrators because of the obvious potential for spamming.

Of course, if you are using the list server to send out mailshots to customers or even to staff members, it would be an excellent idea from a public relations viewpoint firstly to be as polite as possible, and secondly to make it as easy as possible for people to unsubscribe from the list (assuming that this is appropriate for the list in question).

## Sorting

A mailing list is not a useful collaborative working tool if all that happens on the users' desktops is that mailing list traffic appears in the inbox along with all other mail. It's therefore imperative to configure mail clients so that mail from a mailing list is automatically sorted into a separate folder, and then arranged by conversation or thread rather than being jumbled together. Of course, if the mailing list is just used to send the occasional marketing message, technical bulletin or human resources update then this won't be such an issue, but will be vital for busy discussion groups.

Users can readily sort incoming mail from a mailing list by using the From address. NTLlist also supports an alternative method that might be more obvious to the user, which is to include the list name within square brackets in the subject line of each message.

In an internal environment, if everyone has the same email client it's easy enough to include step-by-step instructions on how to set up a new folder and an accompanying sorting rule within the first message sent to new joiners. If users have different mail clients, NTLlist would allow you,

---

*“Implementing a list server that still uses a traditional unsubscribe interface is likely to result in persistent calls to the helpdesk from frustrated users.”*

---

through its ODBC interface, to send different instructions to users of different software, assuming that the information was available in a database.

When setting up sorting rules, it's generally a good idea to configure the software in such a way that any autoreponder message that the user sets up - for example, to inform people who send them email that they are away on holiday - does not respond to mailing list mail. There's nothing more annoying for other mailing list members than to see autoresponder holiday messages.

## Archives

When joining a collaborative list for the first time, users may wish to see an archive of messages from the recent past. All list servers offer this facility, but with varying degrees of usability. For example, some will send the archive as a series of attachments - which could run into thousands - to an email message. Trying to read these is an

awkward, frustrating and time-consuming process for the user, and is thus quite impractical.

List servers increasingly offer the ability to read archives over the Web, which is much easier. Some list servers will carry out the necessary conversion on the fly, in response to a user demand, while others require that the administrator regularly prepares and publishes html archives. This is a trade-off between administrator workload and list server burden.

## Administration

Although the task of joining and leaving mailing lists is the responsibility of the user, there is still a fair amount of administrative work involved in running a list server. A careful study of the work involved is essential, because the burden, and hence cost of ownership, can vary considerably between the list server products available.

The administrator interface is usually either an email-only interface or a Web interface, with the latter becoming much more popular. A Web interface is much more productive when it comes to such activities as moderating a list, which could require on-screen editing of messages before passing them on to the list, and this is a facility provided by Lyris.

## Installation

A wide choice of mail servers is available for both Windows NT and Unix, although not many are available in versions for both platforms. NTMail, for example, is obviously an NT-only product, and the more traditional products such as Majordomo and ListProc are available for Unix platforms alone.

Installing and configuring a list server is usually straightforward, although most administrators today prefer those such as NTMail that have a Web-based interface to those that require knowledge of Unix commands.

Some list servers are supplied with a companion mail server, and some - but not all - are suitable for use with an external mail server. Both Lyris and NTMail, for example, can either be used with the internal mail server or an existing mail server such as Microsoft Exchange.

When configuring a list server for an external mail server, the amount of work will vary according to the server in use. For a standards-based server you just need to specify how the list server collects and sends mail - usually, from a POP3 account and via an SMTP server respectively - and how frequently it should check for new mail. For interfacing with Microsoft Exchange, the process is somewhat more complicated.

When configuring a list server it will usually be necessary to set up at least one administrator account, and further accounts for people who are going to manage lists on the server. The very nature of mailing lists is such that you don't have to set up and administer a user database, since users take charge of adding and removing themselves from the lists, but there is usually some means of importing lists from external sources.



Figure 1 - NTList sports a Web interface to facilitate list server administration.

# Mailing Lists

It's also usually possible for an administrator to add users manually, and of course this will be necessary where mailing lists are private and membership needs to be restricted. With NTList, because it can use an external database you can actually send mailshots to an existing list of users without first having to create separate user lists within NTList itself.

It might be thought that an LDAP interface would be helpful in order to allow a list server to obtain user addresses from an existing enterprise directory. However, when running a mailing list on the Internet with public users this would be of little use, since the enterprise would not already have a directory that holds the details of potential subscribers. For internal use, an interface such as ODBC or SQL, as found in NTMail and Lyris respectively, addresses the ability to retrieve user details from an existing data source.

## Performance

For performance and stability reasons when handling the high volume of mail that can arise from running

mailing lists - sending 50 messages a day to each of 10,000 users comes to 500,000 messages in total, for example - it can be better to use an integrated companion mail server, rather than an external product. Both NTList and Lyris offer this capability.

Of course, you may not want to handle this volume of mail as quickly as possible, because that may swamp your bandwidth and reduce Internet performance for other users. NTList includes the ability to restrict the volume of mail that goes out by inserting a delay between each outgoing message.

One design aspect of the list server that will impact performance and scalability is the way it stores user names. Lyris can use an external SQL database such as Oracle, which offers the ultimate in scalability, and NTMail provides the choice of using either its own internal high-performance database or an external database via ODBC.

## Error Handling

The biggest administrative burden is likely to come from error handling. Although users can leave the mailing

list themselves, if external users change email addresses or internal users leave the company or move departments, then their email address may no longer work. Mail sent to them will result in errors.

List servers usually attempt to handle errors automatically, with varying degrees of success, and this is an important factor since the remaining problems - which could amount to half or more of all errors received - will have to be dealt with manually.

One of the best techniques is that used by Lyris - and is to always populate the standard error return clause of the mail header. Since Lyris sends a unique message to each user, any error message sent back by the recipient's mail server can be identified as belonging to that user, who can then be removed from the list automatically. Only in a very tiny percentage of cases - probably about 1% - will non-compliant mail servers fail to return errors to the correct address, defeating this technique.

## Attachments

You may wish to remove attachments sent to the list, or limit their size in addition to checking for viruses, so check that the list server does allow you to do this. For mailshots it may be better not to send attachments at all, since their bandwidth impact could be horrendous. A more common practice these days is to make the file available on a Web or ftp server and include the URL within the message instead.

## Mail Loops

One of the problems that list servers suffer from is mail loops. If a user is away on holiday and sets up an autoresponder, it will generate a holiday message. When this gets posted to the list it will cause another message to be sent to that user, which in turn will generate another holiday message, and so on.

There are many ways to overcome such mail loops, both manual and automatic. Of course, if a list is moderated, the moderator will delete holiday messages and the loop will never start in the first place. Another technique

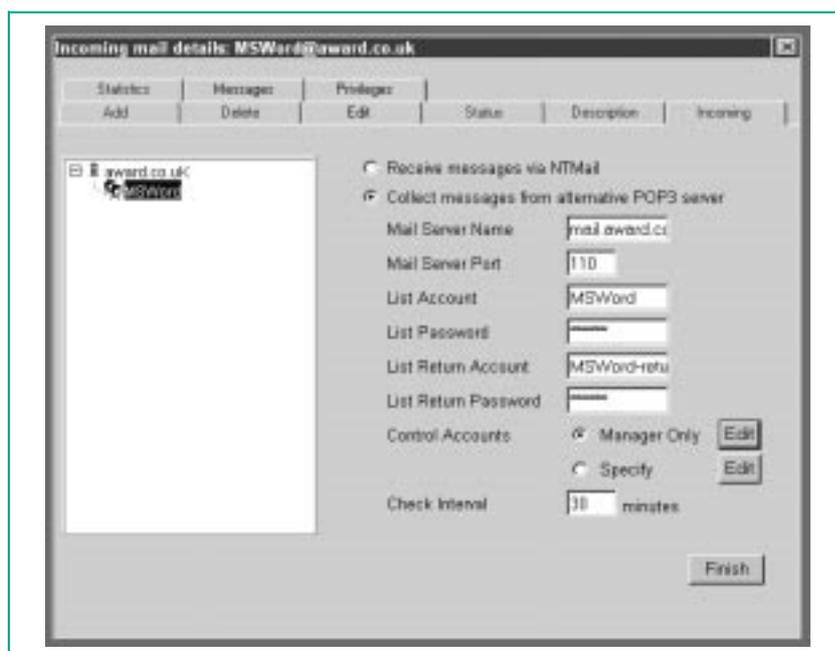


Figure 2 - With NTList, you can specify the mail server to be used to fetch incoming messages on a per-list basis.

often used by list servers to prevent both mail loops and spamming is to implement a posting confirmation process, which stops short of full moderation. Messages sent to the list are placed in quarantine, pending confirmation from an administrator that they may be sent out to the list members. Obviously, holiday messages would not be approved.

Automatic methods can involve checking for words such as holiday or vacation within the message body, or testing to see if a user sends the same message twice in one day. Other techniques involve tracking the number of messages sent in one day, but setting the right threshold is tricky. It can be quite legitimate for a user to send 20 different messages in a day, but that's far more holiday messages than the other subscribers would wish to see.

Another security technique that indirectly prevents mail loops is to require users to include a password for access to the list within the body of each message posted to the list. Any messages lacking the password are refused by the mail server. However, this technique doesn't help usability, and one can envisage frustrated users who forget this requirement - or the password - complaining that their messages aren't being posted to the list.

### Preparation

List servers may make it easy to send out a marketing or technical bulletin to many thousands of users, but they don't help when it comes to dealing with the replies. If you send a mailshot to 600,000 people, it's a good idea to be prepared to handle the response effectively, since there will be a huge peak in replies around two to three days later. The same should be borne in mind when sending mail to internal users, whether it's related to a new pension opportunity that will result in an inundated human resources department or informing them about a technical issue that might lead to a deluge of support calls.

The ability that NTMail has to insert delays between outgoing messages can be particularly useful, since it can achieve spreading of a mailshot over several days, and thus responses will

be spread out too. It also helps to think about timing - starting to send a business mailshot last thing on a Friday that will take three days to send won't help much, since it will just result in a flood of calls on the Monday (if your weekend is Saturday and Sunday).

Attachment size will of course make a huge difference to the speed at which a mailshot goes out. Sending a 2 MB patch file to all registered users of a product may seem like a good idea, but will have a severe impact on your Internet bandwidth.

### Virus Scanning

According to the 1999 Computer Virus Prevalence Survey, email is now the most likely source of virus infection, so anyone operating a mailing list - especially one that has external users - would be very wise to guard against either receiving or transmitting viruses via this medium.

There are many ways to achieve this. If the list server and mail server can't themselves interface directly with a virus scanner (NTMail can), other alternatives include using the firewall, which may have the ability to either carry out scans itself or to use an external scanner. A further possibility is to configure the firewall to route all mail traffic to a separate server, which then scans mail attachments before passing them to the mail server.

### Message Customisation

Customisation of the messages themselves can be quite important when used for newsletter traffic. For years now we have been used to mailmerge software that will automatically change salutations and other aspects of printed letters, and it's natural to expect the same facilities in the email world. NTList, for example, allows unlimited customisation by virtue of its scripting language MML (Mail Meta Language).

Customisation can be a useful tool in many circumstances, whether used internally or externally, or for technical or marketing purposes. Cross-selling is an obvious marketing use, perhaps to inform purchasers of red handbags that you now also have a

line in red shoes. Customisation according to technical criteria, such as the make of video card installed in the PC, can be used to tailor the content of the advice or technical bulletin being disseminated.

### Conclusion

Reliability and scalability are important factors to take into consideration when choosing a list server, but probably more important is the flexibility and cost of ownership. Ease of management will be of particular interest to anyone likely to be involved in its administration, and a Web-based interface makes this task a great deal easier.

Used with care, list servers can be a very effective tool in a number of different areas, from providing technical support within an organisation to creating a sense of community among customers or users of a public Web site. Because they avoid the requirement to use a new product or user interface, user acceptability can be good.

### Web References

A list of mailing servers is at:  
[serverwatch.internet.com/listservers.html](http://serverwatch.internet.com/listservers.html)  
 Lyris: [www.lyris.com](http://www.lyris.com)  
 Gordano: [www.ntmail.co.uk](http://www.ntmail.co.uk)

PCNA

### The Author

Andrew Ward is a freelance journalist and writer who specialises in business and technology subjects. He can be contacted as [andrew.ward@itp-journals.com](mailto:andrew.ward@itp-journals.com).

## New Reviews from [Tech Support Alert](#)

### [Anti-Trojan Software Reviews](#)

A detailed review of six of the best anti trojan software programs. Two products were impressive with a clear gap between these and other contenders in their ability to detect and remove dangerous modern trojans.

### [Inkjet Printer Cartridge Suppliers](#)

Everyone gets inundated by hundreds of ads for inkjet printer cartridges, all claiming to be the cheapest or best. But which vendor do you believe? Our editors decided to put them to the test by anonymously buying printer cartridges and testing them in our office inkjet printers. Many suppliers disappointed but we came up with several web sites that offer good quality cheap inkjet cartridges with impressive customer service.

### [Windows Backup Software](#)

In this review we looked at 18 different backup software products for home or SOHO use. In the end we could only recommend six though only two were good enough to get our "Editor's Choice" award

### [The 46 Best Freeware Programs](#)

There are many free utilities that perform as well or better than expensive commercial products. Our Editor Ian Richards picks out his selection of the very best freeware programs and he comes up with some real gems.