Email Permission Keys

Adrian E. McElligott
Geobytes, inc.
Brisbane 4503, Australia.
adrian@geobytes.com

Abstract

In this paper we look at the relationship between lost messages (false positives) and lost users (churn) and argue that lost messages are costing email service providers their users. We identify the various types of lost messages and attempt to quantify their characteristics. We look at how users acquire email addresses and introduce a new type of false positive reduction technology called 'Email Permission Keys'. We describe numerous types and forms of keys, their purpose, advantages and disadvantages. Further we describe the requirements of three different types of key issuing facilities and the types of keys typically issued by each. We explore the dynamics of user confidence in systems that utilize user feedback or training, and identify the benefits of supplementing user feedback with an automated "Permission Keys" system. We discuss two different implementation architectures, system scalability, and likely return on investment. Finally we argue that exposing a user to spam in their Spam folder is still exposing the user to Spam, and that the best way to avoid this expose is to avoid false positives.

1 Introduction

During 2003, Ferris Research estimated that the cost of lost messages to the end user was $3.50 per message. However, unfortunately they did not quantify the cost of a lost message to the ISP, the Spam filter provider, or to the web mail service provider.

While the cost to the end user may be significant[1], it is born across hundreds of thousands of users, where as the costs to email service providers is concentrated and born largely by only a few hundred providers. Current "user orientated" research may have only revealed the tip of the Iceberg. As users experience and sophistication increases, so does user awareness of lost messages, and with that the readiness of a user to move to a more favorable solution. Perhaps over time, given the user's existing predisposition to churn, the real cost of lost messages may be large enough to sink companies.

From the user's perspective, if the objective of today's Spam filters is to reduce exposure to Spam, then for most users they have failed. This is because whenever a user checks their Spam folder, they are being exposed to all of the Spam - only the folder name is different. Only when a system's false positive rate is reduced to the point where the user no longer checks their Spam folder, is Spam exposure reduced.

The point is that if the user is routinely checking their Spam folder then the filter is of diminished value, and is more likely to be swapped out for a more promising solution.

Gartner (2003) estimates that 7% of ISP Churn is directly attributed to spam.

So why are users still checking their Spam folders when the anti-spam industry is spouting such low false positive rates? The answer is false positives, false positives[2], and the fear of a false positive. While the industry's false positive rates may sounds impressive, the user is more affected by the "Lost Message Rate" which is not so inspiring.

2 New Term - Lost Message Rate (LMR)

Surprisingly the industry does not appear to have a term for the *percentage of **legitimate** messages* that are mistaken for Spam. For the purposes of this paper we will refer to this ratio as the 'Lost Message Rate'.

[1] Ferris Research (2003) estimates that blocked legitimate email, or false positives, cost U.S. businesses roughly $3.5 billion in 2003 alone.

[2] Repeated for emphasis.

$$\text{Lost Message Rate} = \frac{\text{Lost Messages}}{\text{Total Legitimate Messages}}$$

2.1 Usage

The Lost Message Rate as an indicator is more reflective of the users experience then that of the internal workings of the Spam filter, and is therefore typically used to answer questions of user behavior such as "What is the user's tolerance to lost messages?" or "How does the lost message rate relate to client churn?".

Lost Message Rate can also be used to quantify things that the false positive rate can't, such as a filters false positive effectiveness for different types of messages. For example the lost message rate for first contact messages would be likely to be different to that of newsletters, which would be likely to be different again to that of replies that are a part of an existing conversation.

The term lost message rate can be applied to a class of message to determine a filter's effectiveness with a given type of message. On the other hand, the term false positive rate would not be suitable for this purpose as it conveys no knowledge of the proportion of the given class of message within the sample being tested.

3 Different Types of Lost Messages

When considering the cost of a lost message, it is useful to classify the messages in to groups. For the purpose of further discussion, let's tabulate the characteristics of each group.

3.1 First Contact

As the name suggests, a first contact message is the message of first contact with the protected user from a given sender. Of the different types of lost messages, first contact lost messages are the least likely to be discovered and manually recovered.

First Contact messages often bring new business or new opportunities. Typically the injury to the protected user would be greater and the user would be less tolerant to lost "First Contact" messages. If discovered, these are the types of lost messages that users remember, that instigate support calls, and that people talk about.

Table 2: First Contact

Feature	Value
Churn Influence	High
User Injury	High
Avoid-ability	Difficult

3.2 Replies

Reply Messages are part of a conversion or the continuation of an existing relationship. They are the easiest lost message to avoid, and bear a high level of user discovery. Fear of this type of lost message is the most common reason for a user to frequent their Spam folder.

Table 3: Replies

Feature	Value
Churn Influence	Medium
User Injury	Medium
Avoid-ability	Easy

3.3 Solicited Bulk Email

This group includes email publications, Newsletter Subscriptions, Automated Confirmation Messages, Validation and Activation emails, and messages from email based services such as Lyris' Content Checker[3], or Google's Alerts[4] service. They are often difficult to avoid as often the same message is sent to multiple users, some of which may incorrectly report the message as Spam. Accordingly, systems that utilize user feedback may suffer heavily from this type of lost message.

Table 3: Solicited Bulk Email

Feature	Value
Churn Influence	Annoyance
User Injury	Low
Avoid-ability	Difficult

4 Email Permission Keys

Email Permission Keys are a unique code or key that is embedded in to an email address in such a way that it is likely to be retained during normal use of that address, and is therefore available to be extracted at a later date when that email address is used to send an email to the protected user (that owns that email address). Permission Keys work with the existing Internet infrastructure that is in place today, and

[3] Lyris' Content Checker (http://www.lyris.com/contentchecker) tells you how your e-zine ranks in Spam Assassin

[4] Google Alerts are an automated email service containing Google search results based on a predefined query or topic. http://www.google.com/alerts

requires no modification to existing third party processes.

4.1 CaseKeys

CaseKeys are a type of email permission key that use the CAsE of the LeTTerS that make up an email address to embed a unique key into every instance of that email address, whether it is obtained from a web site, a newsgroup posting, or the reply address of an outgoing email. A typical CaseKey might look like this: joHN.SmiTH@eXamPLE.Com

4.2 Display Name Annexing

Display Name Annexing (DNA) – is a type of email permission key that appends or encodes a unique key within the Display Name portion of the email address. A typical display name key may look something like this: "John Smith 12345" <john.smith@example.com> where 12345 is the key.

4.3 Plus Addressing

Plus Addressing, (or Minus Addressing) is appending a key to the local part of an email address via standard plus (or minus) addressing. Plus addressing is most appropriate for 'typed-in addresses' – where the email address is to be published on an off-line medium such as a business card, or is provided over the phone. Plus Addressing alone is not recommended for use with on-line services due to non-universal compatibility with some email systems and the proliferation of non-RFC compliant email addresses validation routines. Plus (or minus) addressing is only available where the underlying email system supports it. Gmail for example supports plus addressing, while yahoo supports minus addressing[5].

A typical Plus Addressing key may look something like this: john.smith+12345@example.com where 12345 is the key.

4.4 DNA/CaseKey Hybrid

A DNA/CaseKey Hybrid Key is a combination of the first two methods above. It is a DNA key with a CaseKeyed representation of the protected user's email address included in both the Display Name part of the email address and the "addr-spec address"[6]. It may look like this –

"John Smith (joHN.SmiTH@eXamPLE.Com)" joHN.SmiTH@eXamPLE.Com

It is typically automatically inserted in to all instances of the protected user's email address in all out-going messages.

4.5 Plus Addressing/CaseKey Hybrid

A Plus Addressing/CaseKey Hybrid Key, as the name suggests is a combination of the Plus Addressing and CaseKey methods. It is essentially a Plus Addressing Key that has been CaseKey encoded. It may look like this –

joHN.SmiTH+12345@eXamPLE.Com

It is typically manually issued to a user via a user interface for use on web forms. The idea is that if the form does not support plus addressing then the tag after the plus sign should be removed – thereby falling back to the CaseKeyed representation.

5 How Permission Keys Work

Permission Keys work by providing information to the Spam Filtering Engine on how the email address was acquired by the sender. As mentioned previously, this is achieved by embedding a unique code, or key at the point of email address acquisition, and then checking any incoming messages for a valid key, thereby ensuring that they are not mistakenly miss-classified as Spam.

While it is advantageous for as many issued email addresses as possible to contain a valid permission key, it is not necessary for all instance or issued email addresses to contain a valid key. Conceptually, permission keys are only used to prevent a false positive, and the absence of a key should not increase a message's likelihood of being classified as Spam.

As we have seen in the proceeding passage, permission keys take many forms, but the underling purpose is to carry a unique code or key which can be used to connect incoming messages with issuing events.

The reason why there are multiple forms of permission keys is to optimize the likelihood of preservation of the key while presenting the email address in a form compatible with the intended method of acquisition.

6 How email addresses are acquired by the sender.

Everyone has an email address, but do they have your address? If so, then how did they get it? Unlike your house keys, you give your email address out to everyone. You put it on your business card, you email it to a friend, who forwards it to a mutual friend, but

[5] Using various separators between the base name and tag are supported by several email services, including Runbox (plus and minus), Google Mail (plus), Yahoo! Mail Plus (minus), and FastMail (plus) http://en.wikipedia.org/wiki/E-mail_address#Plus_.28or_Minus.29_addressing

[6] "Normally, a mailbox is comprised of two parts: (1) an optional display name that indicates the name of the recipient (which could be a person or a system) that could be displayed to the user of a mail application, and (2) an addr-spec address enclosed in angle brackets ("<" and ">"). " http://www.faqs.org/rfcs/rfc2822.html

typically and most frequently you distribute it insentiencely - as it goes out in every message that you send.

There are of course other ways that people acquire your email address. Following is a brief summary of each method along with the preferred form of permission key that is best suited for distribution via that method.

6.1 Acquired from a received email

The most common method whereby people acquire an email address is from an email that they receive, either directly from the protected user, or via a third party. Typically one of the first things that a user does when they change their email address is to email it to all of their friends. Once received, the email address will generally be added to the recipient's address book.

Preferred form of key: DNA/CaseKey Hybrid Key

6.2 Acquired from a web page

While it is fairly uncommon now days to see an email address published on a web page, many users still prefer to send an email then to fill out a web form. Accordingly, the mailto tag is still featured on many sites, and the mailto tag still accounts for a significant number of 'first contact' messages.

Preferred form of key: Plus Addressing/CaseKey Hybrid Key, if the protected user's mail system supports it, otherwise a CaseKey

6.3 Acquired via a web form

The second most popular acquisition method is via a web form. This includes signing up for social networks, email publications, newsletter Subscriptions, and other services where an email address is required to be entered in to a web form.

Preferred form of key: Plus Addressing/CaseKey Hybrid Key, if the protected user's mail system supports it, otherwise a CaseKey

6.4 Acquired off-line, via phone, business card, etc

Typically off-line acquisition involves the sender manually typing in the protected user's email address. For example: from a business card, over the phone, or from off-line media. Direct "typing in" of an email address is error prone and unreliable - so users tend to avoid it. Accordingly the proportion of first contact messages that contain manually typed in email addresses is very low.

Preferred form of key: Plus Addressing Key, if the protected user's mail system supports it.

7 Permission Key Issuing Facilities

New Permission Keys are randomly generated and issued from a key issuing facility. The embedded codes of new keys are recorded along with the details of their provisioning. These details may include: issuing facility, key form, time, and expiry date, and if known, who the key was issued to.

It can be seen from Figure 1 that there are three different types of key issuing facilities, each capable of issuing different forms of keys, and each positioned to reduce one or more types of false positives.

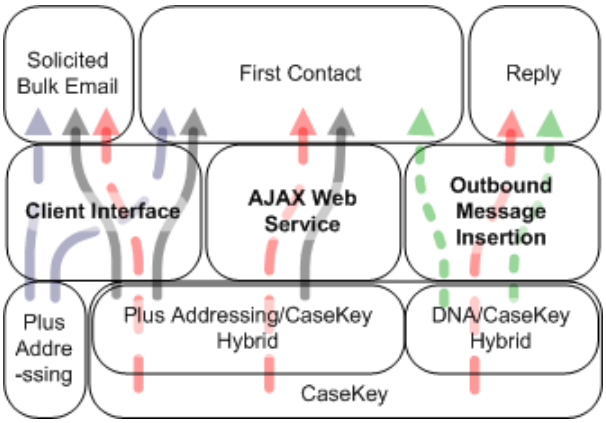

Figure 1: Each type of issuing facility targets one or more particular types of false positives.

The types of key issuing facilities are designed to cater for each of the acquisition methods that were described previously, and are as follows:

- Outbound Message Insertion,
- AJAX Web Service, and
- Manual Key Issuing Facility (Client Interface)

7.1 Outbound Message Insertion

The purpose of the Outbound Message Insertion module is to intercept all out going messages and to embed permission keys in to all instances of the protected user's email address.

Usually implemented at the out bound SMTP gateway, its implementation is relatively easy and simple, requiring no user interaction, and no user interface.

Typically inserts DNA/CaseKey Hybrid Keys, but if high speed in place stream insertion is required, then may insert only CaseKeys.

7.2 AJAX Web Service

The purpose of the AJAX Web Service Key Issuing module is to dynamically insert a permission keyed instance of the protected user's email address in to the contents of a web page – typically within a mailto tag.

While it may be unadvisable to publish an email address on a web page, users still do it. If the user must publish their email address then at least by inserting a permission key we can guarantee delivery of legitimate messages that flow from it. This service also gives us the opportunity to include obfuscate techniques to at least make the address as difficult as possible to harvest.

Permission Keys that are published on web pages should be set to "auto-expire" - we recommend setting them to auto-expire after 7 days. The AJAX service should automatically cycle the Permission Key for a web site on a daily basis - one new unique key would be issued each day. This allows the site visitor 7 days to use the CaseKey before it expires.

This works because the nature of usage of an "on-line published" email address is that the address will be used at the time of issue. A user will click a "mailto:" link and typically send the message within a few days at the most. However, Spambots take time to harvest messages, sell the lists, and finally send the spam sometime later, by which time the CaseKey has expired.

Permission Keys do not block Spam, they detect false positives. Even if the user were to keep the CaseKeyed email address and to use it after it had expired then their message would be no worse off than it was sent prior to implementation of the Permission Keys system. On the other hand however, Permission Keys technology will ensure that for users who do send before the Key Expires, that their message will not be mistaken for spam.

Note: In user feedback dependant systems Permission Keys that are set to auto expire should be excluded from 'is not Spam' voting and their use should be limited to ensuring that a message is not placed in the user's Spam folder.

Typically issues Plus Addressing/CaseKey Hybrid Keys, if the protected user's mail system supports it, otherwise a CaseKey.

7.3 Client Interface

A manual ad-hoc permission key issuing facility is required to facilitate the use of permission key embedded email addresses with Web Forms and for off-line uses – such as printing on business cards.

Ideally such a facility will allow the user to indicate the purpose of the key, and will issue the address encoded in an appropriate form for the indicated purpose.

8 How Permission Keys Help

Permission Keys are used to match an incoming message with an email address issuing event. This information can then be used by Spam Filters to identify legitimate messages that may have otherwise been mistaken for Spam, and thereby improve the filter's false positive rate.

In systems that benefit from user feedback, Permission Keys automate the "Is not Spam" button – which helps in two ways.

- Firstly, it identifies messages that would otherwise be false positives, and
- Secondly, it provides the "feedback" required to dynamically train the filter in real-time.

One problem with user feedback systems, particularly successful ones, is that their users may not check their Spam folders very often and this can result in the effectiveness of the filter degrading. Permission Keys address this issue by providing timely automated feedback.

8.1 How Permission Keys Reduce 'First Contact' Lost Messages

Typically "first contact email addresses" are acquired via either, a web page, a web form, an email, or typed-in from a business card. As described above Permission Key Issuing Facilities are provided to cater for each of these acquisition methods. Specifically, these facilities are as follows:

- Outbound Message Insertion,
- An AJAX Web Service, and
- Manual Key Issuing Facility

8.2 How Permission Keys Reduce 'Reply' Lost Messages

Replies are the most common form of email message and fortunately "Reply Lost Messages" are the easiest to avoid.

The Outbound Message Insertion module as described above intercepts all out going messages and embeds permission keys in to all instances of the protected user's email address.

The DNA/CaseKey Hybrid Keys that are inserted are particularly robust and are typically preserved when the recipient replies from either the original recipient address or an alternative address, when added to an

address book, or when forwarded for use by a third party.

For example, if the protected user sends an email to sales@example.com and the reply is sent from john.smith@example.com then the permission key would typically be preserved.

8.3 How Permission Keys Reduce 'Solicited Bulk Email' Lost Messages

As described above, a client interface is provided to facilitate the ad-hoc issuing of Permission Key embedded instances of the protected user's email address in a form that is optimal for pasting in to a web form. If the protected user's email facility supports plus or minus addressing then a Plus Addressing/CaseKey Hybrid key may be issued, subject to the user being aware that in the event that the web form does not allow plus addressing, that the Plus Addressing Tag may need to be removed.

Alternatively, the default behavior may be to issue a plain CaseKey only, with an option for the more sophisticated user to select a "Plus Addressing/CaseKey Hybrid" key as required.

9 How Permission Keys Enhance User Feedback Dependent Systems

9.1 User Trust oscillation

While user feedback dependant systems have proven to be very effective in identifying spam, they are limited by their user's tolerance to lost messages.

With reference to Figure 1 below, the blue line represents the system's lost message rate in a typical user feedback dependant system, while the dotted orange line represents the user's false positive tolerance threshold. As can be seen from the graph, the systems lost message rate is bound to the user's false positive tolerance threshold and will oscillate around it.

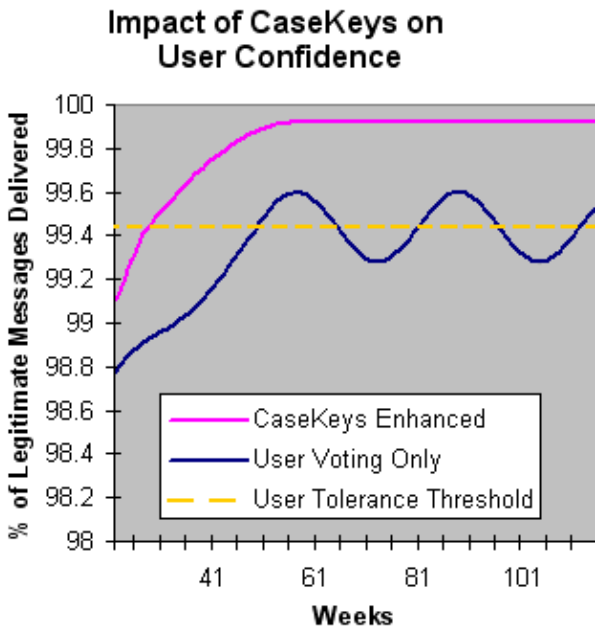

Figure 2: Impact of CaseKeys on User Confidence

This is because, once the system achieves a sufficiently low lost message rate to secure the user's trust, then "Is not Spam" votes plummet, resulting in the degeneration of the system to the point where the user's trust is withdrawn and "Is not Spam" votes return to their previous levels. The cycle then repeats with the unfortunate side effect being that the system fails to exceed the user's tolerance threshold 50% of the time.

Permission Keys enhances the operation of a user feedback dependent system by producing high volumes of high quality "Is not Spam" votes. "Permission Key" votes are of several orders of magnitude larger in volume, and more detailed and reliable than human votes. Permission Key votes identify who the unique key was issued to (the recipient's email address), when it was issued, and where it was issued (in a sent message or web page appearance). This allows the user feedback dependent System to weigh the value of the vote, facilitating an increase in the sophistication of the filter's algorithms. A Permission Key Enhanced System also provides equal treatment for small domains which typically carry a high proportion of business to business messages, and have been historically disadvantaged in the past.

Removing the reliance on the user to identify false positives has other benefits as well - such as dramatically increasing the systems response time, and maintaining a higher level of user satisfaction and confidence. Typically, the lost message rate as indicated by the magenta line in the proceeding graph can be maintained at a level where almost all users cease checking their Spam folder. To these users, Spam is no longer an issue - it is a thing of the past.

10 Invalidating Compromised Keys

Compromised permission keys are invalidated by the 'is Spam button'. Should a key be compromised by a Spammer and used to circumvent the Spam Filter, then the message would appear as a false negative in the user's inbox. Should this occur then the user indicates the message is Spam via the 'is Spam button' in the usual way and the compromised key is automatically invalidated.

11 Implementation Architecture

While actual implementation will vary depending on the existing email system's architecture, conceptually a permission keys system works by inserting keys at one or more insertion points, and then monitoring the existing filter's "Spam Determinations" for the presence of valid keys and recovering any false positives as necessary.

11.1 Minimal Implementation

A minimal implementation requires only two components - an Outbound Message Key Insertion Module, and a Spam Folder Monitoring Module.

Such a system will issue only a single key form – "DNA/CaseKey Hybrid" keys, and will provide no user interface. This 'basic' system runs independent of the user and apart from satisfying users' curiosity there is no user training required.

The advantage of a 'basic' system is that it is very quick, easy, and inexpensive to implement. On the other hand the disadvantage is that it primarily only addresses 'Reply' false positives, and to a lesser extend a significantly smaller proportion of 'First Contact' false positives, with the expected recovery of 'Solicited Bulk Email' false positives to be negligible.

11.2 Full Implementation

A full implementation requires each of the following functional components:

- Outbound Message Key Insertion Module,
- Spam Folder Monitoring Module,
- An AJAX Web Service, and
- Manual Key Issuing Facility.

The advantage of a 'full implementation' is that it reduces a very high proportion of all types of false positive, including 'First Contact' false positives, which are arguably the most challenging type of false positive to avoid, and are also that which causes the user the most pain. In addition, a full implementation is likely to reduce the lost message rate to a point where the user no longer routinely checks their spam folder, thereby greatly increasing the value of the filter to the user.

12 Scalability

12.1 Bandwidth

When implemented according to the preferred configuration where DNA/CaseKey Hybrid Keys are being inserted in to each outbound message, then the permission key adds approximately 50 bytes per message.

Alternatively, when implemented as "CaseKey insertion only", then zero extra bytes are required.

12.2 Data Storage

While the specific requirements of each implementation may vary, the following may be used as a rough provisioning guide.

- Keys per User: 50
- Data per Key: 50 Bytes
- Storage per User: 2.5KB (50x50B)
- Total Storage per Million Users: 2.5GB (2.5KBx1M)

12.3 Data Storage Contention

Although the degree of database contention is largely implementation dependant, database contention should be minimal for the following reasons:

- Only incoming messages that both contain a key and that have been classified as Spam result in a database read request,
- The proportion of outgoing messages to incoming messages is very low, and the vast majority of outgoing messages will only result in a database read request, and
- While 'Initial Contact' outgoing messages do generate a database write, the proportion of 'Initial Contact' messages is minimal as the vast majority of messages will be to recipients that have already been issued a unique key.

13 Early Return on Investment

Initially, all incoming messages will contain email addresses that were acquired prior to the implementation of the permission keys system. Typically the first incoming messages to contain valid permission keys will be replies to message that were sent after the permission key system was first activated, so reduction of 'Reply' lost messages is almost immediate.

‘First Contact’ lost messages are also almost immediately eliminated as the protected user utilizes the key issuing facilities described earlier.

Finally ‘Solicited Bulk Email’ lost messages reduce over time as the protected user utilizes the key issuing facility to register for social networks, email publications, newsletters, and other services where an email address is required.

14 Conclusion

The practical value of a Spam Filter is in its ability to reduce a user's exposure to Spam. Only when a user's confidence is maintained at a sufficiently high level that they no longer check their Spam folder, is their exposure to Spam eliminated.

Users check their Spam folder - exposing themselves to all of the Spam because of their fear of false positives. Each time that they find a false positive, their fear is justified and their confidence falls.

Email Permission Keys can be used to match an incoming message with an email address issuing event. This information can then be used by Spam Filters to identify legitimate messages that may have otherwise been mistaken for Spam, and thereby improve the filter's false positive rate.

Email Permission Keys reduce all classes of false positives. “First Contact”, “Reply”, and “Solicited Bulk Email” false positives can all be almost entirely eliminated.

For systems that benefit from user feedback, the "keys" provide the “is not Spam” feedback in place of the user, allowing the system to maintain user confidence.

Currently, all Spam filters suffer from false positives. Whatever a filter's false positive rate, the integration of a Permission Keys System will reduce it further by identifying messages that would otherwise be routed to the Spam folder.

References

Ferris Research (2007) *Industry Statistics*
http://www.ferris.com/research-library/industry-statistics/

Davey Winder *Hunt for lost email takes five million hours*
Published: Oct 7th, 2007
http://www.daniweb.com/blogs/entry1722.html

Ferris Research (2003) *Industry Statistics*
http://www.ferris.com/research-library/industry-statistics/

Gartner Group (2003)
http://www.gartner.com/

Wikipedia, *E-mail address, Plus (or Minus) addressing*
http://en.wikipedia.org/wiki/E-mail_address#Plus_.28or_Minus.29_addressing.