Problem Patterns of Electronic Messaging Communication

OMITTED RECIPIENT FROM EMAIL MESSAGE
PLoP 2009 – Submission v1.1

Maurice Rabb
PhD Student in Computer Science
University of Illinois at Urbana-Champaign
Research@MauriceRabb.com

Context for PLoP

Since its invention in the late 1960’s, email has quickly risen as the most important, and ubiquitous means of asynchronous electronic communication. It is the lifeblood of communication in most business and professional organizations. However in spite of the importance of email, and the countless person-years spent developing improvements, it still has many fundamental problems. Ironically, for very experienced users of email, these problems are not readily recognizable. Habitual users develop unnamed, yet common methods for handling common email misuse and miscommunication problems.

One email’s greatest strengths is that it is immutable and durable. Unfortunately, this strength is also one of its most nettlesome weaknesses. Its immutability prevents users from correcting common mistakes in a direct and ideal fashion. As a result users have developed conventions and heuristics for repairing these mistakes. Somewhat amazingly, there has been little effort to recognize, understand and classify these common misusage and miscommunication patterns. As part of my research, I am compiling a taxonomy of patterns, with the intent of developing new messaging systems that resolve these problems with more ideal solutions.

In a break with the current pattern conventions, instead of using a single pattern name to encompass both a problem and its solution, I give both a problem and a solution distinct pattern names. This is justified, because most email problem patterns can be solved by several distinct solutions; and many of these same solutions can be applied to solve different email problem patterns.
OMITTED RECIPIENT

One of the most common errors in email is, after sending a message to group of people, realizing that you have forgotten to include one or more recipients.

What is it?

After an email message has already been sent, one of the participants in the message realizes, that one or more additional recipients should have been included in the conversation. The participant who makes the realization and then attempts to repair the omission is known as the fixer. The most basic version of this circumstance is when the original sender is the fixer, and the fix is attempted before any of the recipients have replied.¹

Basic Scenario

Audrey sends Barry, Chris and Felix an email message. At some point (before any recipient has responded), Audrey realizes that she should have included David and Ella as recipients as well.

Why is it a problem?

Users frequently make mistakes in composing email messages. Also, it is not always possible for someone to know a priori everyone who should be included in a message before it is sent. Once a message has been sent, it usually impossible for it to be undone, or otherwise changed.

What are the consequences?

To repair the omission, a separate email message must be sent, including the omitted recipients. If and when the original recipient(s) respond to the original message, the conversation will become disjoint between the two sets of recipients, thwarting the sender's intent of keeping everyone in the same conversation.

¹ Additional clarifying scenarios are discussed in a companion document.
Why does it occur?

It is important to classify the omission of recipients as a problem, and not simply a mistake, because while the problem frequently occurs because the sender makes a mistake, it can also occur due to information that is simply not available at the time the message is sent. In the context of this pattern, mistakes are a subset of possible problems.

<table>
<thead>
<tr>
<th>Source Reason</th>
<th>Problem type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 The sender forgot.</td>
<td>- Memory mistake</td>
</tr>
<tr>
<td>2 The sender didn't know until after the message was sent.</td>
<td>- Missing information</td>
</tr>
<tr>
<td>3 The sender uses a previous message as the basis for the new message, where the previous message was missing one or more recipients.</td>
<td>- Cognitive - HCI</td>
</tr>
</tbody>
</table>

What is the degree of seriousness?

It depends on the context and circumstances. Sometimes is not a big deal, other times it could be a catastrophic social or business faux pas. The problem tends to be exacerbated the more original recipients are involved, and the more messages occur before the omission is realized. The problem is complicated if there are multiple waves of inclusion of omitted recipients. However, the number of newly included recipients (who were previously omitted) within a single wave doesn’t end to increase the difficulty, beyond the fact the conversations with more people are inherently more complicated.

How can it be prevented?

This problem is usually unpreventable, however there are approaches to mitigate against it. One can use other support tools including address groups, mailing lists, or enhanced email client application. See the Related Solutions section below for further discussion.

Standard Solutions

Ideally, a user would like to be able to retroactively add additional recipients to the original email, before the other recipients have a chance to affected by the problem. Since that is not possible is most email systems, users will typically solve the problem in an ad-hoc manner, by either 1) resending the original message with the additional recipients aka RESEND ORIGINAL; or 2) forward the original message to the omitted recipients aka FORWARD ORIGINAL. While RESEND ORIGINAL is usually the superior
solution, FORWARD ORIGINAL has advantages over the former in specific circumstances. Both solutions are workable using all current email client applications.

RESEND ORIGINAL

Before anyone else responds, the sender resends the original message, include the original recipients and adding the additional recipients.

This solution is preferred when it is critical to the sender that the additional recipients remain unaware that they were initially omitted, as the additional recipients will only receive one email.

Ideally, the sender realizes the omission before any of the original recipients have even read the first message. When this is the case, the original recipients will see two identical subject lines in their inbox. Seeing additional recipients in the header fields of the second message, most experienced users will realize the sender’s implicit effort to repair the omitted recipient problem. The recipient will then ignore or delete the first email, and when replying, reply-all to the fixed message.

Given the volume of email that most users receive, it is easy for an original recipient to miss the fixed message; particularly the more time that passes between the original and fixed messages, and especially if the recipient is not using a threaded client app. If it is critical that the original recipients be alerted about the fix, the sender can prepend an annotation to the subject line of the fixed method e.g. “[Resent]”, “[Corrected]”, “[Ignore previous]”, “[This one!]”. Also, this will prevent the original recipients from thinking the fixed message is simply a spurious copy of the original message. However, it must be noted that this will alert the additional recipients to the fix as well.

Further attention to the fix by prepending an explanation to the beginning of the message body about the inclusion of omitted recipients. This is particular useful if the sender wants to ensure that the additional recipients are also made aware that they were omitted from the previous email. However, given a choice between providing a fix alert to the subject line or message body, the former is preferred as it more immediately obvious, when scanning messages in one’s mailbox. This is critical because even if an original recipient fully understands the sender’s intent in fixing an omission, it is still possible for the recipient to accidentally reply to the wrong email, subverting the sender’s effort. Annotating the subject line minimizes this possibility.

It is trickier if the sender wants to ensure the additional recipients remain unaware of the fix, while at the same time alerting the original recipients about the fix. As originally discussed, the sender should send the (un-annotated) fixed message to the all participants, and then perform the alerting of the fix in yet another separate message to the original recipients. Ideally, this alerting message should be sent before the fixing message so that no mistakes are made on the part of the participants. Alternatively, the sender could alert
them using a less formal communications channel such as voice, or via text or instant messaging.

**Differentiating recipient groups**

The more recipients that are in the original message, the harder it may be for a participant to distinguish that additional recipients were added in the second email. This problem is exacerbated, if there are further follow-on fixer emails with additional recipients. Fortunately, if the “cc” field has not otherwise being used, the original and additional participants can be distinguished by putting each group in either the “to” or “cc” fields.

**FORWARD ORIGINAL**

*The sender forwards the original message to just the additional recipients, being careful to include the header info from the original message so the additional recipients can see the addresses of the original recipients.*

This solution is preferred when it is important for the sender *not to resend the original message to the original recipients*, because either:

1) The sender wants to avoid “spamming” the original recipients with the same content; or
2) The original message contained attachments, and the sender wants to avoid the bother and confusion of them receiving multiple documents; or
3) It is critical to the sender that the original recipients remain unaware of the omitted recipients.

Looking at the forwarded message, most experienced users will understand that the sender’s in attempting to alert them to a message in which they were not participants.

However, generally, this solution is more problematic than RESEND ORIGINAL, because though the additional recipients are privy to all the intended message participants, the original recipients are completely unaware of the inclusion of the additional participants. When either the original recipients or the additional recipients reply, it is up to the original sender manage communication between these disjoint groups of participants in the same conversation. At some point, the original sender can create an opportunity to politely merge the two groups together in a single follow-on reply, and finally fully solving the initial omission.

If secrecy about the omission is critical, when the sender forwards the original message to the additional recipients, it is recommended that he/she prepend an instruction to this effect to the body of the message. Alternatively, as in RESEND ORIGINAL, the sender could provide this alert using a less formal communications channel such as voice, or via text or instant messaging.
If the knowledge of the omission is not an issue, when the additional recipients reply, (as above) it is up to them to manually include the original recipients in their reply to re-join the conversation. Likewise, it is useful for the sender to prepend an instruction to the additional recipients to let them know that it is proper (and expected) for them to reply to the conversation.

Related Solutions

Address groups

When a user repeatedly emails the same set of recipients, it is useful to use their client app to create an address group for them. Using an address group makes it easy to add and remove people from a continuing discussion group, and prevents their accidental omission.

Mailing lists

If the group is large enough, going through the trouble of starting a listserv is warranted. Mail the group is as easy as mailing a single identity i.e. the address of the mailing list. Furthermore it is straightforward for participants to review previous messages that occurred before their inclusion in the group, and for them to add or delete themselves on their own.

Alternative Solutions

While current email clients easily support both the RESEND ORIGINAL and FORWARD ORIGINAL solutions, neither are fully satisfying in solving the OMITTED RECIPIENT problem. Neither can guarantee that the complete set of participants will be maintained in follow-on messages to the fixed message. With RESEND ORIGINAL, the sender has no way of ensuring that other participants will response to the fixed message as opposed to the original message. In practice, this accident happens with notable frequency. FORWARD ORIGINAL has the fundamental problem that it doesn’t join the two groups of recipients. It remains, the responsibility of the recipients to manually re-join the participants in follow-on messages. In practice, it is easy for recipients to forget to make the repair; or to be confused about whether it was the sender’s intent for them to take action to join the conversation participants, or simply that the message was meant for them as an FYI.

These challenges are significant enough, that in recent years, other electronic messaging systems have been developed to address some of email’s shortcomings. More flexible messaging system’s such as 37 Signals’ Basecamp and Google’s Wave enable project
groups to avoid the OMITTED RECIPIENT problem entirely. Unfortunately, they require that all participants must fully buy in to using these separate messaging systems in order to enjoy their full benefits. In practice, this isn’t possible for most email users. Even if an entire company moves to an advanced messaging communications system, there will always be other outside people with whom its employees need to communicate. For the foreseeable future, the method will be via email.

Developing systematic ways to better address email problem patterns (e.g. OMITTED RECIPIENT) which demand the minimum change to existing infrastructure and user behavior, will remain an open area of research. Given its persistence and ubiquity, email remains the de-facto lowest common denominator of written electronic communication.

Acknowledgements

This paper was improved through conversations and feedback from Nicholas Chen, Munawar Hafiz, Ralph Johnson, Jeffrey Overbey and Audrey Petty. I would also like to thank Kyle Brown for his thoughtful and patient shepherding of this paper for PLoP 2009.
References


