13

Etiquette-Based Sociotechnical Design

BRIAN WHITWORTH AND TONG LIU

Contents

13.1 Introduction

This chapter argues that etiquette applies not just to the people who *use* computers but also to how the technology is *designed*. When technical systems underlie social environments, as in chat, social networks, instant messages, virtual worlds, and online markets, they must support social acts that give synergy, not antisocial acts that destroy it. Politeness, defined as the *free giving of choice to others in social interactions* (Whitworth, 2005), or etiquette, are critical examples. If we don't consider community good when designing socio-technical systems, we should not be surprised if we don't get it. While etiquette hasn't traditionally been taught in software design courses, we argue that it should be. E-mail spam illustrates what happens when sociotechnical design goes wrong, when it ignores the basic principles of conversation etiquette. Channel e-mail design illustrates an etiquette-based design in the illustrative case of e-mail, but the principles proposed are generic to all sociotechnical contexts.

13.1.1 E-mail Spam

E-mail is perhaps the Internet's primal communication mechanism, being both one of its earliest and most commonly used forms. Yet almost since its inception, it has faced a tide of *spam*—unwanted e-mails from people seeking personal gain. While most of us see spam as a personal inbox problem, the real problem is a community one. Even if all users succeeded in filtering all *their* spam to *their* trashcans, the Internet we share would still have to transmit, process, and store this electronic garbage. The spam your filter "catches" has already wasted Internet resources, and indeed has already been downloaded, processed, and stored by you.

At first, spam seemed more a nuisance than a problem, but in 2003 transmitted spam exceeded nonspam for the first time (Vaughan-Nichols, 2003). So an Internet service provider (ISP) that had one e-mail server for its customers effectively needed to buy another just for the spam. In 2003 over 40% of inbound mail was deleted at the server side by major e-mail providers (Taylor, 2003), though AOL estimated 80% of its incoming 1.5 to 1.9 billion messages a day were filtered as spam (Davidson, 2003). Now spam is the number one

unwanted network intrusion, before viruses, and has always been the number one e-mail user complaint.

While *inbox spam* has remained relatively stable, due to spam filter defenses, *transmitted spam* grew from 20% to 40% over 2002/2003 (Weiss, 2003), to 2004 estimates of 60–70% (Boutin, 2004). In 2006 about 86.7% of the 342 billion e-mails sent per year were spam (MAAWG, 2006), and 2007 estimates are as high as 92% (Metrics, 2006). Since transmitted spam consumes processing, bandwidth and storage whether users see it or not, this is the problem as spam rates increase. While, thanks to filters, many users now find spam a tolerable inbox discomfort, the community problem is growing. Image spam now bypasses text filters, botnets now harvest Website e-mails, and spammers now use real user e-mails as "zombie" spam sources. Historically, transmitted spam is an e-mail problem that has never stopped growing, so it is a problem that won't be going away anytime soon. Our 2004 prediction that within a decade spam transmission rates will rise above 95% unless something changes seems to be coming true all too soon (Whitworth and Whitworth, 2004).

The worldwide costs of spam to people and machines are staggering, and probably underrated. A 2004 estimate of $1934 per employee year did not include IT staff costs, or hardware, software, and bandwidth wasted by spam (Nucleus Research, 2004). A 2003 estimate of lost productivity for U.S. companies was $10 billion (Bekker, 2003), with a 2005 estimate at $50 billion globally and rising (Ferris Research, 2005).

Why has the human community created a technically advanced communication system where most of the messages are created by one computer then deleted by another shortly after, "untouched by human eye"? An alien viewing our e-mail system would suppose its main function was to transmit messages from one machine to another, rather than from one person to another. The system was designed when the social Internet was still just a dream, to efficiently transmit information not meaning, but today we should know better.

It has been argued that spam is an old social problem in new technology clothes (Whitworth and Whitworth, 2004), essentially an electronic "tragedy of the commons" (Hardin, 1968). In the latter example, a village destroys its commons when everyone seeks individual gain and ignores public good. The common communication "field" we call e-mail is becoming a semantic wasteland in the same

way. ISPs operating on limited budgets pay for the Internet bandwidth that spammers consume. Their choices are to let paying customers put up with slower Internet access, to absorb the cost of increasing capacity to pay for the spam, or to raise customer rates. That the spammers who cause the problem pay nothing for its solution is, however, socially unsustainable.

This chapter argues that *sociotechnical problems,* like spam, are technical manifestations of social problems. Hence, they need technology enabled social solutions, i.e., sociotechnical solutions. This chapter is not about etiquette for e-mail users, which is covered elsewhere (see http://www.e-mailreplies.com/), but about reducing "rude by design" software (Cooper, 1999). E-mail, spam, and channel e-mail illustrate a sociotechnical system, sociotechnical problem, and sociotechnical solution, respectively.

13.1.2 Technology Responses to Spam

Most current spam responses are technology-based responses that oppose spam by using technology to combat it directly, without considering social issues, e.g., code filters.

13.1.2.1 Get a New E-mail Anyone who uses the Internet exposes themselves to spam. To estimate how e-mails get onto spam lists, in 2005 we created dozens of new Yahoo, Hotmail, Gmail, and university e-mail accounts, then counted the spam response for these different online uses:

1. *No actions:* Do nothing
2. *Select newsletter options:* Select to receive Yahoo, Hotmail, and Gmail news notices.
3. *Register e-mail to online gambling sites*: www.grouplotto.com, www.freestuffcenter.com, www.findgift.com
4. *Register e-mail to join online groups:* jokes_to_make_u_laugh@yahoogroups.com, thick_angels@yahoogroups.com, Laugh_Loudly@yahoogroups.com
5. *Register e-mail to educational sites:* www.ftponline.com, www.sun.com, ibm.com
6. *Use in online shopping*: www.ecost.com, www.tigerdirect.com, www.walmart.com

Table 13.1 Average Weekly Spam By Online Actions

ACTION	SPAM
No actions	0
Newsletter options	0
Online gambling	7,177
Online groups	1,720
Educational sites	52
Online shopping purchase	46

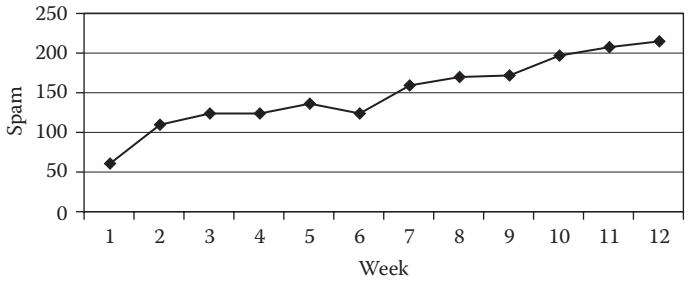

Figure 13.1 Average weekly spam: gambling use.

Table 13.1 shows the total spam for four e-mails over 12 weeks of monitoring. The good news is that while using "risky" online gambling sites gave significant spam, unused new e-mails, even with newsletter options ticked, gave zero spam. E-mails used for online purchases or educational sites gave minimal spam, while online group e-mails gave some spam. The bad news is once an e-mail is listed, spam steadily grows, probably because spammers share lists. In Figure 13.1 while gambling e-mail spam starts at only 50 per week it steadily rises to over 200 per week. For indiscriminate Internet users, the e-mail probably becomes unusable over several years. One solution is to start afresh with a new e-mail, but while this may be financially "free" with Hotmail or Gmail, it is socially expensive as one has to recreate one's social links. Even for cautious e-mail users who only shop online and join groups initially, minimal spam may grow over the years. The long-term effect of spam is to reduce social networks and capital on the Internet.

13.1.2.2 Filters Spam filters, currently the main spam defense, use logic to identify spam content on arrival and put it in the trashcan for

later deletion. As machine learning filters improved, with advanced similarity-matching methods and compression techniques (Goodman, Cormack, and Heckerman, 2007), spammers sent more spam to counter their losses, and found new ways to spam (Cranor and LaMacchia, 1998). When machine learning filters identified spam words like "free," spammers wrote "f-r-e-e," or inserted blank HTML comments like "f<!---->ree" which became "free" when rendered. Spammers can now bypass text detection entirely by sending spam images inside random innocuous e-mails impervious to text filters. If providers develop image-matching filters spammers can randomize image content, or if providers block Web image e-mails, spammers can embed images in the message or disassemble it to be reassembled only when the e-mail is rendered (Goodman et al., 2007). There seems to be no end to this arms race as spam "… is almost impossible to define" (Vaughan-Nichols, 2003, p42). As the "spam wars" advantage shifts back and forth, the only predictable outcome is that transmitted spam will steadily grow. This war is degrading our common communication system. Filters may even exacerbate the problem, as users isolated behind filter walls are unaware of the rising spam flood.

Filtering before transmission could reduce transmitted spam but has the unintended consequence of hiding filter false positives (real e-mail filtered as spam). E-mail filtered at transmission means the sender is not notified (or spammers could tailor spam to the filter), nor is the receiver notified (as no message is sent). So users could never recover real messages categorized as spam, as we currently occasionally do. E-mails with accidental "spam words" would be filtered, and neither sender nor receiver would know, which affects e-mail trust. An apparent e-mail snub could be one's own ISP filtering the outgoing message in secret. Conversely, one could ignore an e-mail, then claim "the filter took it." The postal ethic that "The mail will get through despite hail, sleet, or snow" creates social confidence, which is critical to any communication system, including e-mail.

13.1.2.3 Lists The lists approach uses lists of who is and is not "a spammer" and checks e-mails against them to identify nonspammers (white list) and spammers (black list). However spammers can easily

change identities to avoid black lists, and can "spoof" real users (use them as zombie machines) to get on white ones. Black lists endlessly increase in size, as spammers either create new accounts or spam from a valid account until it is black-listed, then "rinse and repeat" with another account. The long-term sustainability of lists is an issue not just for e-mail but also for all malware, which recently passed the one million known threats mark.

Graylisting uses a combination of black and white lists to reject new e-mails temporarily on the grounds that spammers will move on while real e-mails will try again (Harris, 2003). However, even temporary rejections for 1–4 hours, when messages disappear into an e-mail "limbo," create problems, say, for people awaiting passwords from Web sites.

The administrative effort to create and maintain black/white lists means most individuals don't bother. ISPs maintain such lists, but if one ISP blacklists another, nonspammers in the blocked ISP also have their messages blocked. The logical extension of the list approach is set up a central spam list; for example, in the Tripoli method (Weinstein, 2003) e-mails need an trusted third party's encrypted "not spam" guarantee to be received. Yet if the "trusted third-parties" are institutional bodies, this raises Juvenal's question: "Quis custodiet ipsos custodies?" ("Who watches the watchers?"). Can major e-mail stakeholders like the Direct Marketing Association, Microsoft, or Yahoo define what is and is not spam? Would they not naturally consider their own "useful services" not spam? If we create a universal e-mail gate, whoever controls it could let themselves in and keep competing others out. A central e-mail "custodian" concentrates power, and history suggests that doing this is a mistake.

13.1.2.4 Challenge Responses Challenge defenses or Human Interaction Proofs (HIPs) check if the sender is human by asking questions supposedly easy for people but not computers; for example, MailBlocks asks users to type in the number shown in a graphic (Mailblocks, 2003). That computers can now answer such questions better than people (Goodman et al., 2007) seems less critical than that most spammers never reply to responses (lest they be spammed). The value

of challenges seems to be in the asking itself, not in the question content. Such methods block spam, but e-mail challenges don't save copies, so senders must send e-mail content and any attachments twice. Challenges also increases the psychological cost of sending messages, as senders may take offence at "Are you human?" barriers and not bother.

13.1.3 Social Responses to Spam

Social responses oppose spam with purely social methods aimed at justice. While such methods may use the Internet to find culprits, they do not change computer architectures.

13.1.3.1 Spam the Spammers In simple societies justice is achieved by individuals seeking "an eye for an eye" revenge (Boyd, 1992). The desire for revenge, to punish wrongdoing whatever the cost to oneself, makes antisocial acts less profitable, as long-term vendetta costs cancel out short-term cheating gains. In Axelrod's prisoner's dilemma tournament the most successful program was TIT-FOR-TAT, which always began by cooperating, but if the other party defected then it did likewise (Axelrod, 1984). In the past, revenge "ethics" may have served a useful social purpose, and Lessig once suggested an Internet bounty on spammers, "like … in the Old West" (Weiss, 2003). The method works for companies who fax annoying unsolicited messages, as users can "bomb" them with return faxes, shutting down their fax machines. However, as Internet spammers usually don't accept replies, spam counterattacks go nowhere (Cranor and LaMacchia, 1998), so revenge methods don't work on the Internet (Held, 1998). Also, revenge has negative side effects—for example, in an online vigilante society, a false Internet rumor could shut down an honest company.

13.1.3.2 Laws Large modern societies bypass vendettas by using "the law," where justice is administered not by individuals but by the state, whose police, courts, and prison or fine sanctions change the social contingencies of antisocial acts. In the jury system the law represents the people to punish antisocial acts. Why not then pass a law against spam on behalf of the online community? This approach does not work for several reasons (Whitworth and deMoor, 2003):

1. Physical laws do not easily transfer online (Burk, 2001; e.g., what is online "trespass"?).
2. Online worlds change faster than laws do (e.g., functions like cookies develop faster than they can be assimilated into law).
3. Online architecture is the law (Mitchell, 1995; e.g., programmers can bypass spam laws as if e-mail senders are anonymous, the justice system cannot identify spammers).
4. Laws are limited by jurisdiction. U.S. law applies to U.S. soil, but cyberspace is not within America, as a global Internet spam can come from any country.

Laws like the U.S. CAN-SPAM (Controlling the Assault of Non-Solicited Pornography and Marketing) Act fail because the Internet is not under the jurisdiction of any country. A country can "nationalize" their Internet, then claim jurisdiction over it, by preventing and monitoring access to "outside" sites or e-mails. China may be moving in that direction by blocking access to controversial entries in sites like Wikipedia or Encyclopaedia Britannica, but such blocking reduces China's information synergy with the rest of the world. For the international Internet to "collapse" into national Internet "tribes" would be a tragedy. Instead of evolving to larger social groups, humanity would be devolving to smaller ones (Diamond, 2005).

On a practical level, legal prosecutions require physical evidence, an accused, and a plaintiff, but spam can begin and end in cyberspace. With easily "spoofed" e-mail sources, and a "plaintiff" that is everyone with an e-mail, what penalties apply when identity is unclear and each individual loses so little? The law still cannot contain *telephone* spam (telemarketing), let alone *computer* spam. Traditional law seems too limited, too slow, and too impotent to deal with the global information challenge of spam. This failure suggests the need for an approach that engages rather than ignores technology.

13.1.3.3 Summary Technology methods alone, like spam filters, create an "arms race" between spam and filters, while purely social methods like the law work but are ineffective in cyberspace. Neither social responses (revenge, laws) nor technical responses (filters, lists, challenges) have stemmed the spam tide. Technology responses ignore the social contingencies that create spam, and social responses

ignore the realities of software architectures that enable spam. It is time to try a sociotechnical approach, which fits technical architectures to social principles.

13.1.4 Sociotechnical Responses

The sociotechnical approach to software design involves two steps:

1. *Analysis.* Analyze what social forms are desirable (e.g., polite conversation).
2. *Design.* Design a technology architecture to support these forms.

Note that enforcing "good" by police-state-style tactics is incompatible with sociotechnical axioms of legitimacy, transparency, freedom and order (Whitworth and Friedman, 2009). The method explicitly defines a desired social process, here a polite conversation, then translates its requirements into a technical design that supports both social and technical needs.

13.1.4.1 An E-mail Charge While filters stop spammers directly, social methods reduce spam indirectly by introducing negative consequences for antisocial acts, like prison or fines. This method works for most, and the few immune to social pressure can be hunted down and isolated in prison by crime units. However an anonymous Internet makes this much more difficult; registering every user opens online society to the risk of takeover or hijack, even if all nations could agree to it.

An e-mail charge, applied to everyone, could let the Internet remain decentralized by following the economic principle that people try to reduce their costs, that is, hit spammers in their pockets. In information terms, every transmission could extract a micro-payment, or all senders could compute a time-costly function trivial for all e-mail, so spammers would find the cost excessive (Dwork and Naor, 1993). Such methods essentially suggest redesigning software to increase e-mail transmission costs. However, e-mail charges also reduce overall usage (Kraut et al., 2002), so stopping spammers by slowing the e-mail flow, whether by unneeded charges or pointless calculations,

seems like burning down your house to prevent break-ins. Also it is politically difficult to justify introducing a new charge for services we already have. A new Internet "toll" would add no new service above those now available, as e-mail already works without charges.

Finally, making the Internet a field of profit opens it to corruption. If senders paid receivers, and each e-mail transferred money, who would administer the system and set the charge rate? Is an administration charge effectively an e-mail tax? If so, who then will "govern" online e-mail? Spam works because no-charge e-mail is easy, which is also why the Internet itself works. Its decentralization, with no one "in charge," is why the Internet has so far largely resisted corrupt take over. The social principle of charging for e-mail contradicts the original principle of e-mail success, namely that fast, easy, and free communication benefits everyone, the principle of *social synergy* lying behind the success of the Internet itself. A solution is needed that reduces spam but still leaves the Internet advantage intact.

13.1.4.2 Analysis Spam works under the following conditions:

1. *Benefit.* With an online sucker born every minute, whatever the pitch, someone always takes the bait.
2. *Cost.* E-mail is so cheap it costs little more to send a million e-mails than to send one.
3. *Risk.* The problems spammers create for others have no consequences for themselves.
4. *Ability.* Spammers do what they do because technology makes it possible.

If the response percentage is always positive (#1), the extra message cost near zero (#2), the consequences zero (#3) and e-mail tools provide the ability (#4), is not spam inevitable? Filters try to remove condition #1, but it is a losing battle. With a billion plus worldwide e-mails and growing, spammers need only one hundred takers per ten million requests to earn a profit (Weiss, 2003). Even with filters 99% successful, which they aren't, a hit rate as low as 0.001% still makes spam profitable. The predicted end-point is spammers targeting all users, giving a system that technically "communicates" but mainly in messages no one reads.

13.1.4.3 The Social Requirements of Technical Communication While the first condition seems an inevitable part of human nature, and the second a desirable technology advantage, the third and fourth seem just plain wrong. Technology shouldn't let spammers create negative consequences for others at no cost to themselves, as it doesn't happen in face-to-face interactions.

On a technical level, e-mail is an information exchange, but on a human level it is a meaning exchange that has a name—a *conversation*. Face-to-face conversations have *etiquette requirements* (e.g., just walking up to strangers and conversing usually produces a negative response). People expect others to introduce themselves first, so the other can decide to converse or not, depending on who you are. One may even ask a friend, "Can we talk?" if they look busy. Likewise filibustering at a town hall meeting will get you ejected, as people are supposed to let others speak. The social principle that *conversations are mutual* is an agreed etiquette, based on the politeness principle that each gives choices to the other.

Technology changes the social dynamics (e.g., telephones let anyone call anyone), creating the telemarketing problem of unknown sellers calling at dinner times, reducing the social capital and synergy of society. So society implemented telemarketing laws, as it did when the postal system similarly let bulk mail companies send out masses of unwanted "junk" mail. While the architecture has changed from postal to telephone to e-mail, the social response is the same: to assert *the right not to communicate*, to be left alone or to remain silent. This right is framed negatively because laws and sanctions work that way, but can be stated positively: that *communication requires receiver consent*. The underlying social principle is freedom, and the reason societies adopt it is that free societies are more productive (Whitworth, 2005).

Such social concepts are subtle, as viewers don't mind noninvasive media like billboards or subway ads that they can ignore. Likewise, television viewers are free to change channels to avoid advertisements, so it is their choice to watch them or not. In contrast online pop-up ads that hijack the current window, and the annoying Mr Clippy who hijacked the user's cursor, allow no choice. Similarly e-mail spam comes whether one wants it or not; there is no choice as to even delete

a spam message one must look at it. It is the forcing of communication that is the problem. While spam may not register as strongly as antisocial acts like stealing, murder, or rape, it is in the same category of social acts where one party forces another do something they don't want to do. Spam is not a victimless crime just because its effects are spread across millions. The current spam plague illustrates what happens when the social right to consent to communicate is ignored by a technology design. In contrast, more recent technologies like online chat give more choice (e.g., one can't join a chat unless current users agree).

By this analysis, the current e-mail architecture that lets senders place messages directly into a receiver's inbox is socially wrong because it gives users the right to *communicate unilaterally*. Instead of giving senders all rights, and receivers no rights at all, e-mail should share the right to communicate between sender and receiver (Whitworth and Whitworth, 2004). In file transfer, the opposite problem occurs; receivers have all rights and senders have none, causing the social problems of plagiarism and piracy. In online communication control should be shared (Duan, Dong, and Gopalan, 2005), as per the following social requirements:

1. All conversations require mutual consent.
2. One has the right to refuse any conversation.
3. Anyone can request to converse with another.
4. Once a conversation is agreed, messages can be exchanged freely, usually in turns.
5. One can exit a conversation at any time.

Channel e-mail is a technical design that meets these requirements by sharing communication control between sender and receiver.

13.1.4.4 Channel E-mail The channel e-mail protocol supports conversation requirements via a conversation "channel," an entity that sits above the messages sent. Instead of managing messages, channel e-mail users manage channels (which are less numerous, as one channel has many messages). A channel's recency is that of its last message, as in Gmail threading. An open channel grants mutual rights to freely send messages between online parties, as in current

e-mail. Only if there is no open channel must one be negotiated, via channel "pings"—small e-mails with permissions. Opening a channel is a separate step from sending a message, like the handshaking of face-to-face conversations and some forms of synchronous communication. The handshaking can be automated, letting users just send their messages while the computers negotiate the permissions.

Instead of the current "send and forget" one-step protocol, channel e-mail has several steps:

1. *Channel request.* A conversation request (A to B).
2. *Channel permission.* A permission to converse (B to A).
3. *Message transmissions.* Conversation messages are transmitted mutually.
4. *Channel closure.* Either party closes the channel.

Messages in step #3 use the channel open permission, so further messages do not need channel requests. Channel control is not just the receiver right to tediously reject e-mails one by one, but the right to close a channel entirely, including all future messages from that source. Aspects of this approach are already in practice. For example, Hotmail recognizes:

1. *Safe senders:* Senders who are granted a channel to send e-mail.
2. *Blocked senders*: Senders who are blocked from sending e-mail (i.e., a closed channel).

DiffMail handles spam by classifying senders into (Duan et al., 2005):

1. *Regular contacts*: Message header and content are sent (pushed) to the receiver inbox.
2. *Known spammers*: Messages are not delivered.
3. *Unclassified*: These messages must be retrieved (pulled) by receivers.

In channel e-mail, the receiver effectively "pulls" a new message by sending a permission to the sender's "push" request. Users can manage channels by setting them as:

1. *Open*. Always accept messages.
2. *Closed*. Always reject messages.
3. *Undefined*. Ask me each time.

Meta-option defaults for new (unknown) channel requests can be set as:

1. *Always accept:* The equivalent of current standard e-mail (i.e., completely public)
2. *Always reject:* Closes off to all unknown e-mails (i.e., completely private)
3. *Ask me:* User decides each time

The operation and feasibility of channel e-mail is now considered.

13.1.4.5 Operation Channel e-mail reduces the number of screen lines to manage as it shows channel threads, each containing many messages, in order of the recency of its last message. In Gmail such *threading* keeps same conversation messages together and avoids flipping between "inbox" and "outbox" to figure out who said what last. Channel e-mail has no inbox or outbox, as it threads solely by sender, not topic. If one sender has many e-mail aliases, they can be designated to the same channel. A Gmail sender who changes a message title starts a new thread, but in channel e-mail changing, the topic doesn't change the conversation thread.

In channel e-mail users manage approved sender channels as they manage their friends in Facebook. All unknown senders, including spammers, go into a separate "Channel Requests" category. Only socially naive software would muddle known and unknown sender messages into one inbox that doesn't discriminate friend from stranger.

Channel e-mail users could manage their channels or just send and receive e-mails as usual with the software handling channel requests (i.e. let the handshaking occur in the background). The user setting "Always accept new channel requests" automatically returns a permission to an unknown sender, which they could then use. A new sender without channel e-mail would receive the permission as an e-mail, explaining that they can reply to this e-mail to use the permission. This minor change seems to permit communications from both spammers and nonspammers, but spam would immediately reduce as spammers almost never reply to e-mail lest they get spammed in return. Conversely, for people it would be a natural etiquette to get permission before sending a first-time message.

Channel e-mail gives receivers a choice over who they talk to by "democratizing" list methods, letting users create personal black and

white lists based on channels. It does not reject ISP controlled lists, but just lets users make their own choices as well. List maintenance, a problem for centralized lists, occurs easily at the local level as every sent message implies an open channel and every rejected message closes a channel. Users naturally opening and closing channels in normal e-mail use effectively define their local lists. Channel e-mail gives e-mail users a choice they should have had in the first place.

13.1.4.6 Feasibility Challenge e-mail systems already use a three-step send-challenge-resend protocol. The challenge "Captcha" question is a task supposedly easy for people but hard for computers, like to read a blurry word. A sender who satisfies the challenge can resend the e-mail. If the three-step challenge protocol can be implemented, so can the request-permit-send protocol of channel e-mail. Yet many find "Are you really a person?" humanity challenges insulting and dislike them. In contrast, in channel e-mail it is *polite* to ask if one can send someone an e-mail for the first time. Real people know that relationships take work.

Channel requests would include sender name, e-mail, and subject but not message content or attachments. These "pings" can be of minimal size, perhaps involving:

1. *Title* (e.g., "Can I talk to you about <topic>? Press reply to receive my message."
2. *Channel properties:* Sender e-mail/name/IP address, receiver e-mail/name/IP address, request date/time, accept date/time.

Permissions could use e-mail properties like sender IP address, and request received date/time or even user defined tags. They are dynamic, so closing and reopening a channel creates new permissions. They could even be tags visible in the subject line, so publishing an e-mail tag like "happyvalley99" on a Web site could let readers paste it into an e-mail title to get a direct channel. This is not designed to be secure but interactive: if it is compromised, one can just reset the permission.

Publishing tags could help classify incoming e-mails (e.g., if class students are given a tag to use for all class e-mails, their messages will automatically sort into a channel, rather than into an already overflowing "inbox" which users must organize manually or by setting complex filters. Channel e-mail users could ask unknown e-mails to use

designated channels: "Please use one of my public e-mail tags in your title: [mycompany], [myname] or [myhobby]." As well as open and close, users could create, delete, merge, and transfer channels. Design options like group channels and public key channels are beyond what can be outlined here.

13.1.5 Theoretical Evaluation

While a spammed network may be technically efficient (in bytes/second), it is socially inefficient if most of the messages transmitted are spam. *Social efficiency* (*SE)* can be defined as the proportion of socially useful bytes sent (for a given time period):

$$\text{SE} = \frac{\text{Non-spam bytes sent}}{\text{Total bytes sent}}$$

This ratio is the proportion of network resources used to send socially useful messages. For SE = 100% all network resources are sending non-spam messages, while SE = 40% means only 40% of the network capacity transmits useful messages and 60% transmits spam messages deleted by filters. A network is *technically efficient* if it transfers information well, but *socially inefficient* if it mostly sends spam no one reads or wants to read.

Using an average 59 Kb e-mail size (Szajna, 1994), an average 12 Kb spam size (Willams, 2007), and an estimated request size of 0.25 Kb, Figure 13.2 compares the theoretical social efficiency of standard and channel e-mail by *spam rate*—the percentage of all e-mails that are spam. It was done for the "worst case" scenario (for channel e-mail), where *all* senders are new contacts needing channel permissions. As shown, while standard e-mail begins 100% socially efficient, under spam assault its efficiency declines rapidly. In contrast, while channel e-mail begins at less than 100% efficient, it remains relatively stable under spam load.

The size of each spam message also significantly impacts social efficiency, as larger spam messages, such as those with images or attachments, use up more network resources. Figure 13.3 shows how standard e-mail social efficiency degrades rapidly as spam *message size* increases for a spam rate of 80%, again for a worst-case scenario where

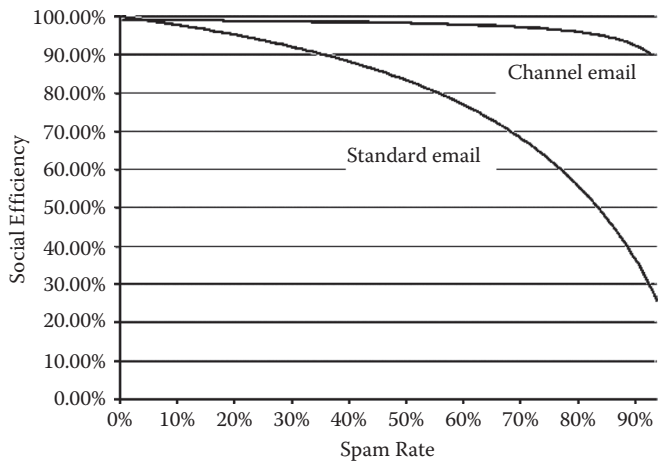

Figure 13.2 Social efficiency by spam rate (theoretical).

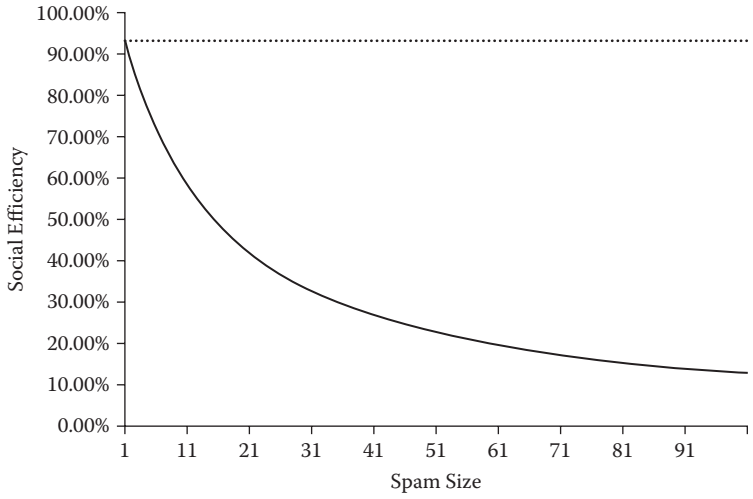

Figure 13.3 Social efficiency by spam message size (theoretical).

all messages are sent by new contacts. In contrast, message size only minimally affects the social efficiency of channel e-mail.

13.1.6 Simulation Evaluation

To verify these theoretical outcomes a channel e-mail communication simulation was set up. One computer sent messages to another over a local network isolated from outside influences, including the Internet. In the standard mode messages were just sent, but in the channel

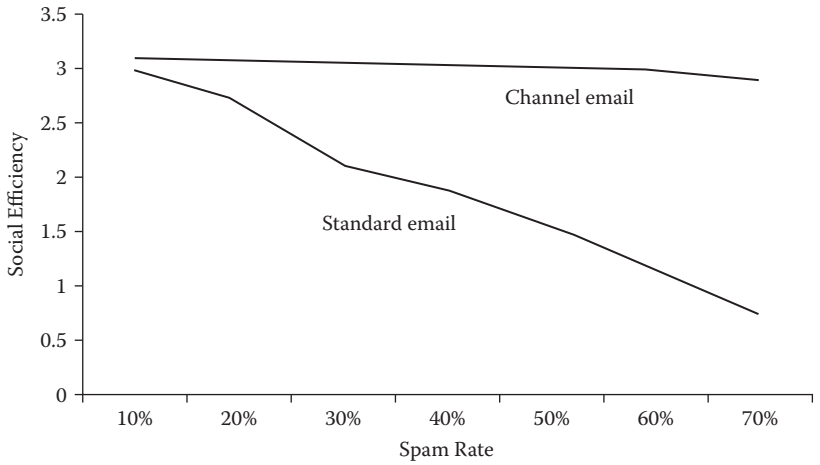

Figure 13.4 Social efficiency by spam rate (actual/simulated).

mode messages required channel permissions. A third computer simulated an outside spam source, using spam message sizes of small (5KB), medium (10KB) and large (60KB), sent at spam rates from 10% to 70% of the non-spam messages. Network social efficiency was estimated by measuring nonspam message *transmission rate* (Mb/sec), which was high if nonspam messages arrived quickly and low if they took a long time due to the spam load. This was taken as a valid measure of social efficiency. Figure 13.4 shows how social efficiency declined by spam rate for medium-sized spam messages. While standard e-mail decreases drastically under spam assault, channel e-mail performance is again robust. Increases in message size also drastically affected standard e-mail but had little impact on channel e-mail.

13.1.7 User Evaluation

Finally, whether user would accept channel e-mail was evaluated by creating Two matching Web-based e-mail prototypes and comparing their usability:

1. *Standard e-mail:* had an inbox with messages received and an outbox with messages sent.
2. *Channel e-mail:* showed channel requests, sent e-mails and current channels. Current channels were divided into open (known), closed (spam) and not yet classified areas.

In the channel prototype e-mails from first time senders went into a channel requests area, where users could press buttons to "Accept sender" (move to open channel area) or to "Reject sender" (move to spam area). It could later be moved again to another area (e.g., to unclassified). This study evaluated user interface acceptance rather than network performance, so the prototype did not use a three-step channel protocol, but just sent simple messages. Subjects were in groups of 10, each with an allocated e-mail ID. Their task was to send 17 simple e-mail questions to other participants, like "What is your birthday?" and also to respond to 17 such questions from other participants. A total of 34 valid e-mails had to be sent and replied to per person, which took on average 43.4 minutes. In addition, incoming spam was sent to all participants at rates of 12%, 40% or 73% of valid e-mails sent. Subjects were divided randomly into two groups, one evaluating the traditional e-mail prototype, and the other the matching channel e-mail prototype. Each group completed the task for each of three spam levels then completed a questionnaire with four usability dimensions: Understandability, Learnability, Operability, and Perceived Usefulness, based on a validated model (Bevan, 1997). The survey questions were:

1. I would find it easy to get this e-mail system to do what I want to do. (Understandability)
2. To learn to operate this e-mail system would be easy for me. (Learnability)
3. I would find this e-mail system to be flexible to interact with. (Operability)
4. Using this e-mail system in my job would enable me to accomplish tasks more quickly. (Perceived Usefulness)
5. Using this e-mail system would make it easier to do what I want to do. (Perceived Usefulness)

The seven-point response scale was from extremely likely to extremely unlikely.

Figure 13.5 summarizes the results, where the channel e-mail interface rated higher than standard e-mail on both usability and usefulness dimensions. A t-test comparison of the mean response scores for questions 1–5 for standard versus channel interfaces was significant at the 0.01 level, suggesting users clearly preferred channel e-mail.

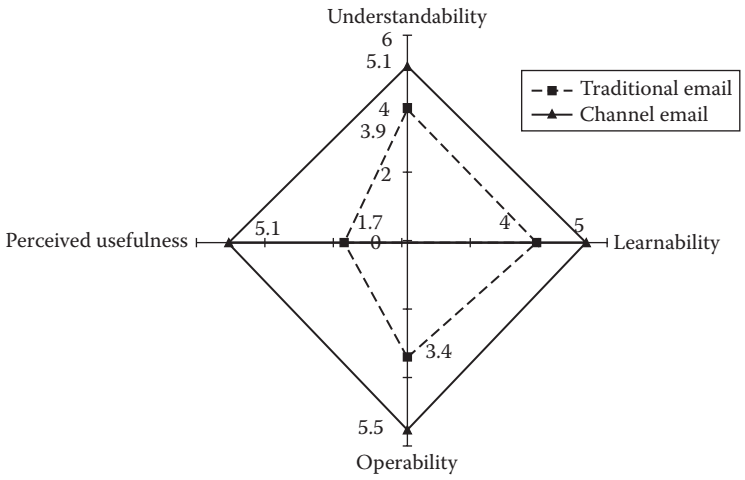

Figure 13.5 A usability comparison of traditional versus channel e-mail.

13.1.8 Implementation

To deploy a new system over an existing one it must be backwards compatible; to grow and develop, it must also be useful.

13.1.8.1 Compatibility For a useful new system to catch on it must first survive an introductory period when only a few people in the community actually use it. To survive this phase, it must be backward compatible with existing, more broadly used systems. Channel e-mail has two such cases:

1. *Non-channel sender*: Channel e-mail can treat e-mails from new senders as channel requests, and reply to accepted senders with a channel permission: "Press Reply to send your e-mail by this channel." Since the sender may not be familiar with the channel idea, it would include a "canned" explanation of what a channel is and why spam makes it necessary.
2. *Non-channel receiver*: In this case, the permit ping; the channel that appears as a first time e-mail to someone else, appears as a polite request: "XYZ wants to send you an e-mail on <topic> Just press reply to receive the e-mail." Again, it could explain channel e-mail, and that this is a once-only connect request. The pending message would automatically be sent on receiving a reply.

Users converting to channel e-mail could minimize the impact on their friends by sending out channel invitation e-mails using their address book. In this system it takes effort to manage one's friends, as cell phone users for example do. Once set up, a channel is easy to use, but spammers must now do work to establish a new communication channel.

13.1.8.2 Usefulness Channel e-mail modifies and combines features from *black/white lists* and *challenge e-mail*.

It democratizes black/white lists, as users can personally define who they talk to by opening and closing channels. We naturally know this, so systems like Facebook succeed by giving users tools to manage their friend/not-friend lists. Spammers can bypass ISP controlled black lists of known spammers by creating a new identity, but as individuals usually work by white lists of friends, a spammer with a new identity is still "unknown."

Channel e-mail has no "Are you human?" challenge, but its three-step process challenges the sender to reply to the permission, which spammers almost never do. *That spammers never reply is the perfect spam filter*. If spammers adopt channel e-mail to blend in, transmitted spam would still be reduced as Figure 13.6 shows. Channel e-mail gives a return on network resources for all spam rates over 10%. At an 80% spam rate it saves about 40.8% percent of network resources, so at current rates it would allow an Internet service provider to replace three e-mail servers by two.

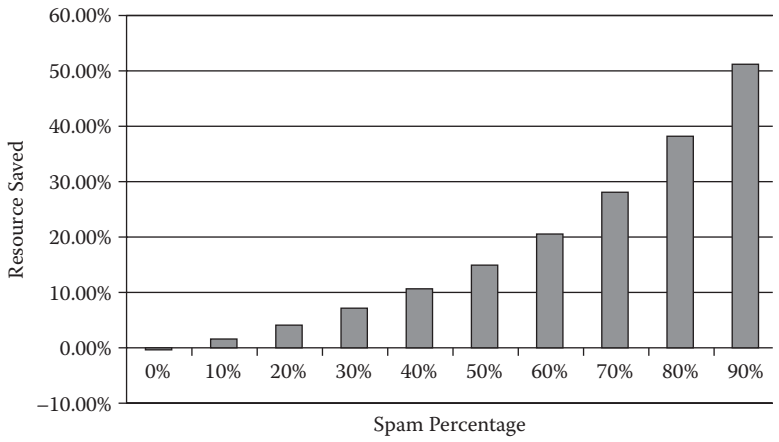

Figure 13.6 Network resources saved by channel e-mail, by spam rate.

13.1.8.3 Organizational Spam Channel e-mail also works with *organizational spam* where both filters and lists fail. While in commercial spam a few people send millions of unwanted messages, organizational spam is many people sending unwanted messages, like: "My daughter needs a piano tutor, can anyone recommend one?" This message was actually sent to everyone in a large organization. While commercial spammers epitomize selfishness, organizational spammers are just ordinary people being inconsiderate. Yet it is as big a problem as commercial spam. When people spam a community list, standard e-mail only lets one unsubscribe, but channel e-mail gives two more options

1. *Return to sender.* While commercial spammers don't receive e-mails, organizational spammers do. The *Return to Sender* button deletes the message and forwards it back to the sender with a "No, thank you" note (Whitworth and Whitworth, 2004). Under channel e-mail, spamming a group list could results in hundreds or thousands of return-to-sender replies.
2. *Close the channel.* Choosing this option automatically creates an e-mail such as "The user has closed this channel. To reopen …" and any further contacts from that person become new channel requests. This doesn't close the list, just that sender.

Channel e-mail lets the community give feedback to those who abuse its goodwill. Since people are often extremely sensitive to social censure, the social effect on those who routinely send unwanted e-mail messages to community lists could be dramatic.

13.1.9 Discussion

The approach taken here differs markedly from Microsoft's "more of the same" approach, whose researchers feel smart filters are "holding the line," and that *we* will defeat *them* in the spam wars (Goodman et al., 2007). Online history doesn't support this view. Indeed, it is predictable that two sides with the same assets, of human cunning and computer power, will inevitably produce an endless spam arms race, and indeed transmitted spam has never dropped. Spam researchers may rejoice that spam wars will be "keeping us busy for a long time to

come" (Goodman et al., 2007), but why use our Internet commons as your battleground?

Simple arithmetic suggests that technology alone cannot contain the spam challenge forever; with over 23 million businesses in America alone, each sending just one unsolicited message per year to all users, there are over 63,000 e-mail messages per person per day. Spam potential grows geometrically with the number of users, and with billions online in the future it easily outstrips Moore's law of technology growth.

If e-mail "dies" from spam overload, don't imagine we can just move to other applications as the spam plague already infects them, e.g. SPIM (IM spam) and SPAT (chat spam). Acting like slash-and-burn farmers is not an option. Spam is just the poster child for a genre of antisocial acts threatening online society, including spyware, phishing, spoofing, scams, identity theft, libel, privacy invasions, piracy, plagiarism, electronic harassment, and other Internet "dark side" examples. Spam is a social problem that online humanity must eventually face.

An ideal world might have no spam but in our world, it is a reality. Yet it needn't overwhelm us, as societies have managed antisocial behaviors for thousands of years. By *social evolution* ideals like legitimacy, democracy, transparency, and freedom have emerged (Whitworth and deMoor, 2003). Politeness is one of these, and as such deserves support.

Spammers are the commercial fishing trawlers of the information world, whose huge technology created nets making previously abundant environments barren. Obviously taking continuously from a physical system without returning anything is not a sustainable process, but it is less obvious that this applies equally to social and informational systems. One option is to introduce fair play rules that limit "catch" sizes. In social systems, the positive path to the same end is to encourage people to give consideration to each other by supporting an etiquette.

Traditional society punishes antisocial acts by sanctions like prison, but the Internet can support social acts by technical design. In "polite computing," code supports beneficial ideals like fairness and democracy; for example, channel e-mail gives senders and receivers equal rights to communicate. Online social rights are defined by what the

computer code allows, which is as if physicists could define the laws of physics in physical space. As we create online social environments, the code that creates the anarchy of spam also allows Orwellian control of all online user expression. On this question, of how our socio-technical environment will be built, software designers cannot sit impartially on the sidelines. Let us embed what physical society has learned socially over the centuries, written often in blood, in socio-technical designs. Principles like freedom, transparency, order, and democracy deserve design support (Whitworth and Friedman, 2009). The role of etiquette and politeness in this is to be a positive force. While *laws* define what people *must* to in legitimate social interactions, *politeness* defines what they *could* do to help each other. While one aims to obstruct "evil," the other aims to direct "good." Sociotechnical software that supports positive social interaction by design is better than trying to chase down and punish negative interactions.

Hence, channel e-mail doesn't try to find or punish spammers. It doesn't target them at all. Its target is the unfair social environment that "grows" spam. Fair rules apply to all equally, so anyone can choose to converse or not, and any e-mail can be rejected, not just spam. The goal is fair communication, not punishment or revenge. A community that is given the tools to discern and choose between social and antisocial acts will choose what helps it survive. As transmitted spam moves steadily to an equilibrium end-point of over 99%, the value of positive sociotechnical design will become increasingly apparent.

Acknowledgments

Thanks to Deepthi Gottimukkala, Srikanth Kollipari, and Vikram Koganti for the spam data, to Victor Bi for the channel e-mail interface, and to Zheng Dai for network simulations.

References

Axelrod, R. (1984). *The Evolution of Cooperation*. New York: Basic Books.

Bekker, S. (2003). Spam to Cost U.S. Companies $10 Billion in 2003. *ENTNews, October 14*.

Bevan, N. (1997). Quality and usability: A new framework. In A. M. E. van Veenendaal, J (Ed.), *Achieving Software Product Quality*. Netherlands: Tutein Nolthenius.

Boutin, P. (2004, April 19). Can e-mail be saved? *Infoworld,* 41–53.

Boyd, R. (1992). The evolution of reciprocity when conditions vary. In A. H. Harcourt and F. B. M. de Waal (Eds.), *Coalitions and Alliances in Humans And Other Animals*. Oxford: Oxford University Press.

Burk, D. L. (2001). Copyrightable functions and patentable speech. *Communications of the ACM, 44, February*(2), 69–75.

Cooper, A. (1999). *The Inmates are Running the Asylum: Why High Tech Products Drive us Crazy and How to Restore the Sanity.* USA.

AU: Publisher's name?

Cranor, L. F. and LaMacchia, B. A. (1998). Spam! *Communications of the ACM, 41*(8), 74–83.

Davidson, P. (2003, April 17). Facing dip in subscribers, America Online steps up efforts to block spam. *USA Today,* p. 3B.

Diamond, J. (2005). *Collapse: How Societies Choose to Fail or Succeed*. New York: Viking (Penguin Group).

Duan, Z., Dong, Y., and Gopalan, K. (2005). A differentiated message delivery architecture to control spam. Paper presented at the *11th International Conference on Parallel and Distributed Systems (ICPADS'05)*.

AU: Date, venue, publisher's name and place of publication?

Dwork, C., and Naor, M. (Eds.). (1993). *Pricing via Processing or Combatting Junk Mail*. New York: SpringerVerlag.

Ferris Research. (2005). Reducing the $50 Billion Global Spam Bill [Electronic Version]. *Ferris Report*, *February* from http://www.ferris.com/?page_id=73258.

Goodman, J., Cormack, G. V., and Heckerman, D. (2007). Spam and the ongoing battle for the inbox. *Communications of ACM, 50*(2), 25–33.

Hardin, G. (1968). The tragedy of the commons. *Science, 162*, 1243–1248.

Harris, E. (2003). The next step in the spam control war: Greylisting. from http://projects.puremagic.com/greylisting/whitepaper.html.

Held, G. (1998). Spam the spammer. *International Journal of Network Management, 8*, 69–69.

Kraut, R. E., Shyam, S., Morris, J., Telang, R., Filer, D., and Cronin, M. (2002). Markets for Attention: Will Postage for E-mail Help? Paper presented at the CSCW 02, New Orleans.

MAAWG. (2006, June 2006). MAAWG E-mail Metrics Program, First Quarter 2006 Report. Retrieved April 1, 2007, from http://www.maawg.org/about/FINAL_1Q2006_Metrics_Report.pdf.

Mailblocks. (2003). Mailblocks is the ultimate spam-blocking e-mail service.

Metrics. (2006). 2006: The year spam raised its game; 2007 predictions [Electronic Version]. MessageLabs December from http://www.metrics2.com/blog/2006/12/18/2006_the_year_spam_raised_its_game_2007_prediction.html.

Mitchell, W. J. (1995). *City of Bits Space, Place and the Infobahn*. Cambridge, MA: MIT Press.

Nucleus-Research. (2004). Spam: The Serial ROI Killer [Electronic Version], Research Note E50 from http://www.nucleusresearch.com/.

Szajna, B. (1994). How much is information systems research addressing key practitioner concerns? *Database, May*, 49–59.
Taylor, C. (2003). Spam's Big Bang. *Time, June 16,* 50–53.
Vaughan-Nichols, S. J. (2003). Saving Private E-mail. *IEEE Spectrum, 40*(8), 40–44.
Weinstein, L. (2003). Inside risks: Spam wars. *Communications of the ACM, 46*(8), 136.
Weiss, A. (2003). Ending spam's free ride. *netWorker, 7*(2), 18–24.
Whitworth, B. and Friedman, R. (2009). Reinventing academic publishing online Part II: A Socio-technical Vision. *First Monday*, *14*(9).
Whitworth, B. (2005). Polite computing. *Behaviour and Information Technology, 24*(5, September, http://brianwhitworth.com/polite05.pdf), 353–363.
Whitworth, B. and deMoor, A. (2003). Legitimate by design: Towards trusted virtual community environments. *Behaviour and Information Technology, 22*(1), 31–51.
Whitworth, B. and Whitworth, E. (2004). Reducing spam by closing the social-technical gap. *Computer* (October), 38–45.
Willams, I. (2007). Image spam doubles average file size. Retrieved October 10, 2007, from http://www.vnunet.com/articles/print/2186424.