Welcome Visitors

A Pattern Language for Textual Electronic Communications

A work in progress

Email is often described as the killer app for the Internet. People spend more time writing and reading email than any other Internet activity. When one adds in instant messaging, online chats, and other forms of textual electronic communications, the people affected by the design of these systems is enormous. The purpose of this pattern language is to help people design appropriate and humane textual electronic communications into systems they are building.

Using this pattern language we hope will allow system designers to create applications that make use of a range of ways for people to communicate, including traditional email, bulletin boards, chat rooms, instant messaging, web communities, and other ways of textual electronic communications. Each application would use the appropriate methods to meet the human needs of its users. These human needs must address that people are differentially abled/disabled: some have limited or no vision, others have trouble typing or using a mouse, and most are not computer professionals.

Start here:

- PatternLanguageForTextualElectronicCommunications --- our patterns so far
- ExemplaryPatterns --- pointers to some of our better efforts
- HowThesePatternsWereWritten --- or why are there multiple versions of some patterns and none of others
- SequencesOfPatterns --- to be a real pattern language we need sequences too
- LevelsOfScale --- different levels we feel the pattern language needs to address
- JoinUs --- we need your help. NextSteps includes ideas for making progress.

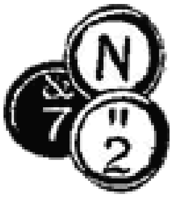
Search this site:

About this site:

- WhosWho
- GoodStyle
- http:changes.cgi -- Recent changes
- http:changes.cgi?max=100 -- More changes

Edit

Last edited December 25, 2002
Return to WelcomeVisitors



About The Logo

The apl4tec logo is a collage of antque typerwriter keycaps. The source images are from Junque, a rubber stamp design and manufacturing company. This particular arrangements is Copyright (c) 2002, Ward Cunningham and used here with permission.

- http://www.homestead.com/prosites-junquegrrl/typewriteralphabets.html
- http://www.homestead.com/prosites-junquegrrl/ordering.html

I am coresponding with Junque regarding the fair use of their images. -- WardCunningham

---

Why N-7-2? The numbers and letters don't mean anything. I just picked images that looked good and didn't cost much. this is what we got. -- WardCunningham

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Ward Cunningham

- http://c2.com/~ward/
- mailto:ward@c2.com

I host the site, encourage the authors and designed the logo. See AboutTheLogo.

Edit

Last edited December 25, 2002
Return to WelcomeVisitors

How These Patterns Were Written

It may be appropriate to say a few words about how these patterns came to be.

After the OOPSLA meeting where we decided to try to create a humane pattern language for email progress was very slow---in fact almost non-existent. So in November 2001 several of us (JoshuaKerievsky, RichardGabriel & RonGoldman) met to see about moving things forwards.

We took the ideas we had for patterns---often just the pattern name---and tried to organize them in related groups. When we were done we had 47 possible patterns.

Since Dick was having good results writing a poem a day, we decided to *lower our expectations* and try to write a pattern a day---or at least several a week. The plan was to quickly try to flesh out our potential patterns. For each possible pattern we each would try to write up something and then later merge our different versions. Hence you will notice that some of the patterns are quite short and incomplete.

Over the next few months we made some good progress, writing several versions of most of our candidate patterns and adding some new ones to the mix. But then we ran out of steam. At OOPSLA 2002 WardCunningham offered to create a wiki for our proto-language & we thought that was a great idea. So what's on this wiki are our initial drafts.

The next step is to complete the unwritten patterns, merge the multiple drafts into more refined versions, add in new patterns that people think appropriate, and try to make the whole into an actual pattern language.

Please JoinUs in this. --- RonGoldman

Edit

Last edited December 18, 2002
Return to WelcomeVisitors

Pattern Language For Textual Electronic Communications

Here is our current list of patterns, somewhat organized into categories. Some of the patterns have one or more versions written up, others are placeholders yet to be written.

A Pattern Language for Textual Electronic Communication

- NoManIsAnIsland

- QualityOfConversations
    - IntimateCommunication
        - SharingFeelings or Being Clear or Communicating Subtlety
        - DramaticEffect or Deep Emotion
    - StylizedLetter
    - PrivateConversation - Encrypted
    - PeopleTalkToPeopleMachinesTalkToMachines
        - WhenOurMachinesActForUs
    - SharingInformation

- GroupDiscussions
    - SmallWorkGroupsComeAndGo
    - NurtureCommunity (was Host and Introductions and Culture)
        - FrequentlyAskedQuestions
    - QualityControl - Moderation

- WorkStyles - how a person organizes their day - shifting tasks - mobility
    - TheContextOfOurLives
        - MultiplePersonalities
        - OnePersonManyMachines (or Electronic Nomads)
        - AwarenessOfPresence
        - TimeSensitiveAndLocationSensitive
            - AwayMessages (or Gone Fishin')
        - PrivacyGradient
        - InterruptionGradient
            - GoAway
        - IntimacyGradient
        - InnerCircle

        - CommunicationDoor
        - NotesForOnesSelf
    - RhythmOfConversation
        - SlowLetter
        - QuickChat
        - TimelyResponse (was Faster)
        - ExpedientResolution
        - ReturnReceipt (was Notify User if Mail has been Read/Scanned)
        - EaseOfShiftingModality, eg phone
        - GentleReminder
    - ConveniencesForFastWork
        - LowOverheadMessages
        - IntelligentFormLetters - Template Letters
        - MailMerge
    - CheaperToUse
    - UniversalAccess

- InformationDisplayAndManagement
    - GroupingOrDisplay
        - ConversationsThreading
        - UnifyEmailAndNews
        - KeepArchives
        - MorningPaper (was Digest)
        - MailFolders
    - MetaInformation
        - AnnotatedMessages
        - FlaggedMessages
    - MailFiltering

- MachineIssues
    - GoodIntegrationWithOtherTools
    - EditingEase
    - EasyToUnsubscribe
    - SafeToUse
    - RosettaStone
    - HelpfulSystem

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

No Man Is An Island

No man is an island --- RonGoldman *28 Nov 2001*

* * *

As we live our lives we need to interact with many other people, both at work and at home. These interactions take many forms, such as face-to-face conversations, group meetings, phone calls and email exchanges.

In today's world many interactions are now remote ones, mediated through our technology. This technology can either enhance or degrade the interaction. Unfortunately technology usually gets in the way, making our interactions with others of poor quality.

Email is often described as the killer app for the Internet. People spend more time writing and reading email than any other internet activity. When one adds in instant messaging, online chats, and other forms of textual electronic communications, the people affected by the design of these systems is enormous. Most people when asked will complain quite vehemently about how poorly these programs suit their needs.

Therefore,

Design programs for textual electronic communications so that they are appropriate and humane. Make them support the rich ways that people actually interact. Each application must use appropriate methods to meet the human needs of its users.

* * *

Build programs that support the human aspects of conversation-QualityOfConversations, that allow for GroupDiscussions, and that recognize that each person has their own WorkStyles for organizing their day as they shift tasks and move about. Be sure to apply the principles of InformationDisplayAndManagement. Also consider how these programs interact with other programs—MachineIssues.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Away Messages

Away Messages --- *JoshuaKerievsky 06 Dec 2001*

... not only must our communications software be TimeSensitiveAndLocationSensitive, it must also know how to intelligently inform folks when we're not around. AwayMessages that are intelligent are never re-sent to the same individual or group---GroupDiscussions.

* * *

It is incredibly annoying to send a message to a group list and have some number of emails send to you from folks you've never heard of telling you how they're not in their office, but they'll be back on such and such a date. Who cares!

Yet most email programs are oblivious to group discussions---they are perfectly content to send an Away Message to anyone and everyone, with no limit on how many to send. So if you write 3 emails and send each of them to one email list, you can get 3 Away Messages from the same individual who is surely enjoying himself on some vacation.

People want to send Away Messages so that when friends or loved ones or colleagues send them emails, they can be polite about not returning those emails. So the intention for using Away Messages is good. But the execution is awful---since email programs simply don't differentiate between those who we care to inform or don't care to inform.

Therefore,

Provide an Away Message Assistant that walks users through the process of setting up intelligent Away Messages. Make this Assistant understand when not to send an Away Message and how to send different Away Messages to different people and groups.

* * *

Integrate Away Messages with chat programs---QuickChat so that someone may can receive a meaningful Away Message when we step away from a chat.

Implement rich, expressive Away Messages using IntelligentFormLetters and make it possible to set Away Messages from anywhere---UniversalAccess.

Gone Fishin' (was Away Messages) --- *RonGoldman 12 Dec 2001*

. . . an aspect of TheContextOfOurLives is that we are not always available to others. Our machines should know when we are present---AwarenessOfPresence---and also when we are not there---TimeSensitiveAndLocationSensitive or we do not want to be disturbed---InterruptionGradient.

* * *

Sometimes we are away from our machines for periods of time and out of touch with those that may be trying to reach us.

As social animals people need to maintain their connections with other individuals. When we cannot reach someone we want to know where they are, what they are doing, if there is another way to reach them and when we might be able to contact them. When we are away we want our friends to know so and not think that we are ignoring their messages.

Nowadays most Americans have an answering machine connected to their phone. When they are out or don't want to answer the phone, their answering machine let's callers know they have reached the correct number and that no one is available to talk to them. When people try to reach us using other technology such as email or chat, our software should be able to let them know that we are away.

At the old Stanford AI Lab people used the finger program to check if other people were there. Finger reported back if they were logged in and if so, when they had last typed on their keyboard. If they were not logged in finger reported when they had last used the computer and then printed their "plan" message that often told when they were likely to be around and if they were away on a trip.

Many email programs have a feature that people can enable when they go on vacation. When set, any incoming email causes a note to be sent saying they are away. Unfortunately most email clients are rather stupid and always send the message even when it is not appropriate. More intelligent programs keep track of who they have sent a message to, and so anyone sending several messages to the absent person will only receive one notification of their being away.

These away messages are inappropriate when they do not take into account who has sent them. We don't want the same away message to go to our close friends, to our colleagues and to complete strangers---let alone to those sending us spam! Our machines can easily distinguish among people trying to reach us and provide them the appropriate information, if any.

When away messages are sent in reply to notes sent to a mailing list it is also usually inappropriate. People posting to a mailing list generally do not care about any specific individual on the list; they are talking to the group as a whole. For some small lists, such as a small work group, that are more personal, an away message might be appropriate.

Therefore,

Make our machines able to intelligently act for us when we are away. Have them send appropriate information to those trying to reach us.

* * *

Let a person indicate who is important to them---InnerCircle---and what sort of message they should be sent.

Mark all messages with some type of meta information so that an email program can distinguish between messages sent by an individual, by a mailing list (and which one), and by a bulk mailing (spam). In that way our machine intermediaries can act appropriately---PeopleTalkToPeopleMachinesTalkToMachines.

Email clients can obtain information from calendar programs to know when and where a person has gone---GoodIntegrationWithOtherTools.

Away Messages --- *RichardGabriel 5 Jan 2002*

. . . textual electronic communications takes place within TheContextOfOurLives, because NoManIsAnIsland. We wish to take heed of when there is cross-entity communication-machines to people and people to machines (PeopleTalkToPeopleMachinesTalkToMachines). We want to ensure a TimelyResponse when people try to get in touch with us. The situation may be complicated because we also engage in QuickChat, for which we desire our software to have an AwarenessOfPresence, and in Workgroups (GroupDiscussions) where the rules of politeness and presence are different from ordinary electronic communications . . .

* * *

We want to be responsive to people trying to communicate with us, but sometimes we are unable to respond quickly-either because we are not near a communication device, are too busy, or simply have stepped away. To be polite, we want our machines to respond to messages in such situations in order to inform correspondents that we are temporarily unable to answer rather than letting our silence appear as rudeness.

On the other hand, we may be subscribed to mailings lists as part of a workgroup in which vacation messages are as unwelcome as spam. Here, rather than appearing polite and considerate by having our machines respond to messages, we will appear rude or even stupid. With commonly used email clients, people are currently forced to set up a complicated set of mail filters, typically by hand, in order to filter out vacation messages, and setting up such messages to recognize such only messages from mailing lists is too difficult if not impossible.

When we are engaged in a chat, we might step away from the computer-to get a cup of coffee, inspect the plumbing, or unexpectedly wash the cat-and would like the computer to notice our general inactivity and respond to an instant message with a stepped-away message.

Therefore,

Arrange for your textual electronic communications system to easily set up intelligent away messages---messages that go to the people you want and not to mailing lists or others for whom such messages might not be welcome.

Away messages should work with chat sessions or instant messaging as well, perhaps as a quick check off that says the respondent is temporarily occupied and perhaps by the machine noticing that the respondent is inactive after some period. This wouldn't be like a vacation message, since the expected delay in response is more likely to be minutes than days or longer.

* * *

Implement rich, expressive Away Messages using IntelligentFormLetters and make it possible to set up Away Messages from anywhere---UniversalAccess.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Awareness Of Presence

Awareness of Presence --- RonGoldman *4 Dec 2001*

. . . the pattern TheContextOfOurLives provides for programs to act appropriately in different contexts. One of the most important aspects of any context is what people are involved.

* * *

People are social creatures. We usually work or play with other people. It is important to us to know who those people are and what they are doing.

When we are physically present with other people we keep track of who is where and doing what. If someone appears to be deep in thought or very busy, then we won't interrupt them. If they have left the room we notice that and don't walk over to talk to their empty desk.

When we interact with people through our technology we currently lose this information. We have no way of knowing whether or not they are there to receive our phone call or email. Nor do we know if they have the time to read it.

The reverse is also true: they have no idea if we are sitting in front of our computer or elsewhere. In fact it has been pointed out that the autoflush toilets in airports nowadays are more aware of you than your high-powered "personal" computer---at least the toilet knows whether you are using it or have left (and it acts accordingly).

Our computers' ignorance of us can be seen by the common scenario of where we start up a long task, such as printing a paper, then go away expecting the work to be finished when we return. Often we come back to find no work done and a dialog box requesting a meaningless confirmation. Had the program known we were gone, a more intelligent outcome may have resulted.

Some systems, such as buddy lists, are starting to provide us with information about which of our friends or colleagues is currently online. We don't yet know what they are doing, but at least we know they are logged in.

The Awarenex project at Sun Labs (http://research.sun.com/features/awarenex/index.html) is a system developed to help members of remote workgroups contact each other. At a glance, it makes it possible to tell if someone is active on their computer keyboard, talking on the phone, or engaged in an instant message conversation. It's a way to virtually peek into someone's office window to see if they are available.

Therefore,

Build into computers and other devices ways to sense if we are present and an indication of how active/busy we are. Provide ways for this information to be shared with others we interact with and utilized by programs we work with.

* * *

We don't want everyone to have the same degree of access to this awareness information---I don't want telemarketers to know when I'm sitting by my phone---PrivacyGradient. The information on how busy we are can be used so our computer knows when it's ok to interrupt us---InterruptionGradient. This information is useful both to other people and to other software---PeopleTalkToPeopleMachinesTalkToMachines and AwayMessages. Awareness of presence over time also shows up in weblogs or bulletin board type discussions where it is important to see who is taking part in the discussion---SmallWorkGroupsComeAndGo.

Awareness of Presence --- *RichardGabriel 6 Jan 2002*

. . . We want to ensure a TimelyResponse when people try to get in touch with us, and we sometimes also want to provide other, non-verbal clues about the way we have composed our messages . . .

* * *

When we are talking to another person, we can tell when he or she stops paying attention. The laws of physics and our senses combine to tell us that the person, let's say, has walked away, or started to talk to someone else, or has simply dozed off.

The physical interaction patterns are part of the communication. If we observe someone thinking for a long time and speaking slowly, then we conclude the person is thinking carefully and speaking cautiously, perhaps to not say the wrong thing, or because the topic is sensitive or difficult. Without such clues, it would be much harder to interact with people who are in close proximity. With letters, we can usually assume that writer took some care, because there is a relatively difficult process of creating it involved: With handwritten letters, there is the tactile effort, and with computer-printed letters, there is the need to do the printing and a strong urge to proofread it that everyone seems to feel.

When we are away from our computers, we may not answer even apparently urgent communications for quite a while. This can make us seem rude, impolite, careless, or insensitive. How we wish to appear when we step away should depend on who is trying to communicate and on our preferences.

Textual electronic communications systems don't provide much in the way of cues about how the communication was composed. Sometimes there are lots of typos, which could mean haste. The passage of time means little unless the response is quick, in which case we know it was quick. The software that assists the composition of a Textual electronic communication knows quite a bit about the interaction of the writer with the software while composing the message.

But a person may wish to use the fact that the recipient cannot see the composition-perhaps to send non-verbal signals as powerful as the words on the page. A passionate letter answered with a short note after a long delay indicates that the passion is not shared and communicates, possibly, a desire for the letter-writer to cool his or her passion.

Therefore,

Take advantage of the computer as surrogate eyes and ears for the recipient of a message. Arrange for the textual electronic communications software to know whether the recipient has stepped away, is busy working on something else without paying attention to received messages, has not been active on the computer all day or for days, or whether he or she seems to have stepped away for a few minutes.

Arrange for the software to keep track of how much time and effort was spent on the response and arrange for it to provide annotations or other sorts of cues for the recipient of the response-was this a quickly composed note, did the writer pause frequently in one passage with perhaps many edits and changes. Make it so that the RhythmOfConversation is available to the software as well, in order for it to better cater to the way the correspondents interact.

* * *

The information noticed by the software can have a deep influence on how to complete DramaticEffect or Deep Emotion, GoAway, TimelyResponse, Workgroups (GroupDiscussions), SmallWorkGroupsComeAndGo, AwayMessages, PrivacyGradient, and InterruptionGradient.

Sharing Feelings

Sharing Feelings or Being Clear or Communicating Subtlety --- *RichardGabriel 4 Dec 2001*

. . . you want to be able to convey emotions because NoManIsAnIsland, and you are trying to wrote an ? IntimateLetter, but even were you to take the time for a SlowLetter you are unable to show your emotions well. In Workgroups (SmallWorkGroupsComeAndGo) there needs to be a way to keep misunderstandings under control . . .

* * *

Every person needs to convey emotions, share emotions. Ideally you should talk to the other person face-to-face or by telephone, but sometimes circumstances prevent this. Tone of voice, facial expressions, body movements and stances—--these are the language even the most inarticulate person learns to convey emotions accurately.

But in words? Conveying emotions well using words takes skill and patience, and even talent. Sometimes even the best writers are unable to convey sincerely what they feel or what they want to say that they feel. For many, the act of trying to convey emotions in words takes them out of those feelings, making it even harder to convey them.

Language is about information, surely, but also about the human condition and spirit. It's our blessing and curse to have to heap so much into language. There are some emotions we are afraid to express directly in front of others: some forms of love, hatred, disgust. It depends on each of us.

Even in a slow letter expressing emotions can be difficult, but especially in short notes or letters when you need to write quickly it is still important to get the emotions right. If, for example, you want to show you are not angry in a hurried letter, this could be hard to do because rapid writing is often curt and sounds to the reading mind like the sentences are cutoff. If the recipient is expecting a long letter and receives a short one, what is she to make of this? If she had a clue to the real emotion behind it....

In the early days of textual electronic communication, misunderstood sarcasm often sparked fierce disputes, and wisecracks too often prompted avalanches of response postings.

Therefore,

Use emoticons (emotional icons) and avatars (graphical, sometimes animated characters) to convey emotions. As an alternative, establish an emotional communication mechanism that is like a meta-statement—perhaps as parenthetical statements.

Emoticons can express only crudely how we feel, but they can be used to avoid communication blunders. This is especially true if you are writing sarcastically—the use of a friendly glyph can convey the sarcasm that the ineptness of overfast composition belies. These are some simple emoticons:

```
 :-)    Expresses happiness, sarcasm, or joke
 :-(    Expresses unhappiness
```

:-]	Expresses jovial happiness
:-[	Expresses despondent unhappiness
:-D	Expresses jovial happiness
:-I	Expresses indifference
:-\	Indicates undecided, confused, or skeptical
:-Q	Expresses confusion
:-S	Expresses incoherence or loss of words
:-@	Expresses shock or screaming
:-O	Indicates surprise, yelling or realization of an error ("uh oh!")

The emoticon was first used in email in 1982 by the computer scientist Scott Fahlman. He was trying to find a more aesthetically pleasing way to convey that a statement or passage was to be taken as a joke. Earlier, the word "joke" was added after such a passage, but it seemed to Fahlman to be a poor way to convey the emotion.

* * *

Build systems that recognize the strengths and weaknesses of people, as well as which fit into the way they work and the tasks they do every day. Carefully observe the failures and breakdowns of our tools and endeavor to improve them over time.

Sharing Feelings or Being Clear or Communicating Subtlety --- *JoshuaKerievsky 07 Jan 2002*

. . . when we communicate face to face, our clothes, our tone of voice, our eyes, our posture communicate on our behalf. How can our communication software be such a rich communication channel?

* * *

The richness of our in-person communications is usually lost when we communicate in text.

Many people love to communicate via text because it is inexpensive (CheaperToUse) , fast (TimelyResponse, ExpedientResolution) and less intrusive than a phone call. The trouble is, since most of us aren't professional writers, our communications often lose their richness when we convert them to plain text.

For many, this loss of communication richness isn't a problem—they simply live with it, doing the best they can to communicate effectively. But others routinely get stung by miscommunications that result from their text, such as a jest that is taken seriously. The miscommunications sometimes trigger absurd or even painful events. For example, a miscommunication on a large, public email forum can lead to a public attack on someone's character (QualityControl — Moderation).

We can't prevent miscommunications from happening, but we can give people easy, convenient tools to help make their text communicate better on their behalf.

Therefore

Provide a full-set of visual text enhancements—such as emoticons, ASCII-art, and digital pictures—to authors of textual communications and make it easy to insert these text enhancements into messages by making them visible on the screen and capable of being dragged and dropped or searched and inserted at the touch of a key. Make it easy to add to one's catalog of visual text enhancements.

* * *

Visual text enhancements may be used to communicate DramaticEffect or Deep Emotion, provide a richer communication experience in StylizedLetter or IntelligentFormLetters, and clearly communicate feelings or preferences during an ExpedientResolution. They need to be available to us every machine (UniversalAccess) and must integrate intelligently with other tools (GoodIntegrationWithOtherTools).

Nonverbal Communication (was Sharing Feelings or Being Clear or Communicating Subtlety) --- *RonGoldman 30 Jan 2002*

. . . when two or more people communicate the QualityOfConversations depends on more than just the words they use. Especially for IntimateCommunication it is important to provide ways for people to convey their feelings.

* * *

When we talk face-to-face with another person we use every cue and nuance we can to help us make sense of the message we receive. We look to the person's facial expression, body posture, tone of voice, and gestures as much as to the words they choose. The nonverbal channel enables us to interpret the verbal message: Are they being sarcastic, angry, joking or....

For conversations over other media we do not have access to all of these cues. Over the telephone we only have tone of voice and the rhythm of their speech. Over the computer we are left with just their words.

For someone hurriedly dashing off a quick email note, how will the recipient distinguish between haste and curtness? In a message from someone we do not know well, how do we differentiate between a joke and an insult? The answer is not very well as demonstrated by the countless disputes and flame wars that are seen in newsgroups and on mailing lists.

Writing clearly is difficult enough. Writing to also convey one's feelings takes skill and talent that not everyone has. Moreover not everyone is comfortable sharing their feelings with others.

What has evolved over the last twenty years in email and newsgroups is the practice of adding parenthetical comments to substitute for the missing nonverbal channel. In a message to show they are joking people would write the word "joke" or "<grin>" or "*chuckle*".

On September 19,1982 the computer scientist Scott Fahlman suggested on one of the CMU bboards that people explicitly label comments not meant to be serious with a :-) glyph. Thus was born the smiley or emoticon (emotional icon). Some common smileys are:

Intimate Communication

Intimate Communication --- *RichardGabriel 6 Dec 2001*

. . . NoManIsAnIsland. People need to express emotions (SharingFeelings) as part of the RhythmOfConversation. Communicating private and intimate thoughts are part of TheContextOfOurLives. People require privacy and confidentiality . . .

* * *

High-tech communication focuses on information content—the words "information content" reflect deep concern about technology and shallow concern about humanity.

Transmitting information from one person to another was one of the first applications of the Arpanet—precursor to the Internet. These early mail programs were designed for the purpose of to move research and military information expediently. Soon, however, it became apparent that the medium was going to be used as a substitute for postal mail and telephone calls. Later these early mail programs were used for collaboration on projects. The natural inclination of people to socialize and share personal correspondence rose to the surface.

Instead of only typing in data from the latest experiment or logistical timetables, people additionally started writing letters, expressing grief and joy, and sharing personal milestones. One finds it hard to speak of the expression of love or sympathy over the loss of a parent as "content."

Therefore,

Endeavor to create a textual electronic communications system which recognizes that people will be communicating intimately, expressing their deepest feelings in words typed in to a machine. Do what you can to make the experience human and humane, make the tools adapted and adaptable to how people will work when expressing their most private thoughts.

* * *

Make it possible for people to write a SlowLetter, express Deep Emotion (DramaticEffect), respecting their privacy (PrivacyGradient, PrivateConversation) and dignity. Make the experience transparent (LowOverheadMessages, EditingEase) and

Low Overhead Messages

Low Overhead Messages --- *RichardGabriel 3 Dec 2001*

. . . you are building a textual electronic communication system for very non-expert computer people . . .

* * *

You want your users to be able to compose and send a message without having to work too hard at the mechanics of sending a message. The goal of a user is to communicate with another person or group not with the computer.

Computer-based communication is too hard to understand for no benefit to a person. Once a person is in the mood to send some sort of message, he or she is not in the mood to deal with the mechanics of the software. Yet, the mechanics need to be dealt with in order to make the communication possible.

Therefore,

Build a system that fills in as much as possible of the mechanics of the message. The header for a message should be as completely filled out as possible from the message being responded to. Addressing to new people might be through auto-completion from the database consisting of all received and sent messages as well as an address book. Responses to messages should include the original message in a form convenient to use and modify. Signatures should be filled in automatically. Make it easy to respond to the sender and all recipients of a message or just to the sender. Making forwarding email effortless by constructing all but the recipients' names.

Rather than making the person go to the computer, make the computer go to the person. A lot of the work of sending a message can be automated, so automate it. The letter writer should rarely if ever have to deal with issues surrounding the network connections.

* * *

This pattern can be assisted by IntelligentFormLetters and possibly SharingFeelings. QuickChat might be related since it deals with switching modes, which is a form of mechanics hiding.

Editing Ease

Editing Ease --- *RichardGabriel 6 Dec 2001*

. . . When people are trying to communicate they want their tools to be natural and non-intrusive (LowOverheadMessages, RhythmOfConversation, TheContextOfOurLives). To many designers this means that the windows, buttons, menus, pop-ups, and dialogue boxes should be well designed and well tested. But it means much more than that because the experience of using the tool extends down to the visual presentation of text on the screen, its formatting, and the editing commands . . .

* * *

The experience of a software tool extends down to the smallest physical detail. The fonts on the screen down to the pixel level, the tint of the electronic paper being typed on, the feel of the keyboard and keys beneath the fingers, the height and angle of the keyboard, and the physical size and smoothness of the pointing device.

One thing often overlooked by designers of interactive software is that fact that people can develop an affinity to specific keyboards, pointing devices, fonts, white-paper color, and editing commands. Many designers use the text editing structure of the underlying platform, meaning the operating system, while the designers of that system have sometimes simplified, reasoning that application writers will supply better editing tools.

Computer hardware is designed for people to plug in their own input devices and monitors, but this is rarely the case for software devices.

Some textual communication systems require people to move their hands off the keyboard to a pointing device or arrow keys to navigate while other systems provide chordal navigation commands that allow the typist to leave their hands in typing position.

Writing textual communications is one of the most commonly done editing tasks, so you should provide the person with his or her most comfortable editing commands and features

Therefore,

Do what you can to make it so that editing and other user-input software is broken into components so that people can use the editing structure of their choice.

Expedient Resolution

Expedient Resolution --- *RichardGabriel 29 Nov 2001*

. . . you have decided to write an email client and perhaps are thinking about adding chat facilities . . .

* * *

Some messages are part of a quick back and forth, possibly in order to get a quick resolution about something like the time of a meeting or the place. There isn't a need for thoughtful discussion or well-thought-out opinions and discursive material.

The rhythm of communication is quick, not slow. The need is to get information back and forth without delay and without extra work. The rhythm is almost like a brainstorming session trying to find questions or new approaches.

Therefore,

Arrange for the textual communication system to support quick replies with short-circuiting of details. If it's technologically feasible, enable the client to arrange a chat or instant-messaging session with the recipient client when this rhythm of exchange is noticed or by a simple control or button.

Auto-completion based on a dictionary and words already seen in the message can sometimes help a person trying to get a message out fast. A palette of icons might be useful if they can be manipulated quickly.

* * *

In cases like this it makes sense for editing-moving around in the text space and entering or altering text-to be as easy and familiar as possible, ideally using the person's most favorite text editing engine. See the pattern EditingEase.

Expedient Exchange (was Expedient Resolution) --- *JoshuaKerievsky 11 Dec 2001*

... you've created tools to account for different speeds and styles of online communication---SlowLetter, QuickChat, TimelyResponse. Now you want to make these tools accessible when they are most needed.

* * *

Using email to quickly resolve a matter is like talking to someone on the phone, but having to endure prolonged periods of silence before hearing each others voices.

This isn't an indictment of email. Email has excellent uses. It just so happens that it isn't very good for conducting quick exchanges, particularly when you need to resolve a matter expediently. Chat programs are much better for such exchanges, but because they aren't integrated with our email software---GoodIntegrationWithOtherTools---we often defer to using email, even when using chat would be a better choice. The real problem is that when we need to communicate, we don't know who is online and who isn't.

Consider the case where two people are online, both of whom have chat and email software, and one of them sends the following message to the other:

```
Mary,
      I think today is the deadline for submitting
to the conference magazine.  What shall we do?
                                --Joe
```

Joe may or may not get a quick response to his email, depending on what Mary is doing. Say Mary is online but doesn't have her email software running because she is doing something else. In that case, Joe won't be getting an immediate response to his question until Mary checks email---but had Mary known that Joe was online, trying to contact her about this conference deadline, she would've gladly interrupted her work (InterruptionGradient) to communicate with Joe.

Or consider the case where Mary is online with her email running but she is deeply engaged in some non-email activity. She sees that email arrived, stops her work, finds Joe's message, responds to it and then gets back to work, repeating the process if or when Joe sends another response. This is a slow way to resolve a matter and has the downside of continually interrupting Mary in her work.

There are more scenarios like this, each of which has upsides and downsides. The real issue, however, is how to enable expedient exchanges with folks who matter to us, when we want to engage in those exchanges. To enable this level of sophistication, we need chat and email software to be integrated and cognizant of who is close to us and who is not and what kinds of interruptions and exchanges are tolerable or not.

Gentle Reminder

Gentle Reminder --- *JoshuaKerievsky 05 Dec 2001*

... you are creating communications software that is sensitive to TheContextOfOurLives and the speeds at which we best communicate---RhythmOfConversation. Such software provides an ideal environment in which to compose thoughtful messages---SlowLetter, or facilitate quick exchanges---QuickChat, TimelyResponse, ExpedientResolution. But what happens when we simply don't have time to interact with anyone at all? Usually, a great deal of silence happens: messages don't get answered and message senders are left wondering if something went wrong, if they offended us or if we just didn't care to respond to them.

Nearly all communications software is oblivious to the content of the messages it sends and receives. Thus, when we're too busy to respond to a thoughtful, carefully written message from a friend or colleague, our software doesn't care. It doesn't remind us to respond to the message and it doesn't care if we come across looking like a jerk because we didn't respond for days or weeks on end.

This happens all the time to busy people! Important messages arrive, but because of extremely busy lives, these messages simply go unanswered for days. The message senders either get used to not having their emails responded to us or they just get annoyed with us. And we can certainly understand their frustration, since they took the time to compose an important message, only to be greeted by silence from our end.

But it doesn't have to be that way. Software doesn't need to know exactly what people are writing to us, but it can know, with a certainty, that some emails will be thoughtful and others will be trivial and that there are different responses appropriate for different kinds of messages. We are really talking about common decency.

For example, if Michael sends me a thoughtful and important email, which clearly took him time to compose---I would like to respond to his message with equal thought and care. But if I don't have time to respond, I need my communications software to help me not be a jerk---which is easy to be when one doesn't have time to respond to emails.

My communications software could help me by a) giving me a way to painless and quickly send Michael a message explaining how busy I am at present and b) remind me every X days that I still haven't responded to Michael's email and c) after noticing that I still haven't responded to Michael after Y number of days or weeks, send Michael an apologetic email and exhort me to call the guy on the phone to explain why I've not had a chance to write.

In other words, when we'd like to be polite in the context of electronic messages but don't have the time to be polite, we need out software to go to bat for us---helping us by sending short responses and prodding us to respond as we would like to respond. Therefore,

Equip our communications software with features that make it easier for us to be polite and responsive when we are too busy to be polite and responsive. Create a "Gentle Reminder" button that, when pressed, reminds us at regular intervals to respond to important emails.

A Gentle Reminder could give us the option to keeping reminding us to write a response to someone, while also offering to send them an automated update about our current work---IntelligentFormLetters---Template Letters. . .

Gentle Reminder --- *RonGoldman 9 Jan 2002*

. . . in TheContextOfOurLives there is a natural ebb and flow of activity that creates a Rhythm to our Conversations (RhythmOfConversation) and causes us to neglect some tasks in order to do others.

* * *

We lead busy lives and lose track of things we really need to do.

Everyone could benefit from a personal assistant who would remind us of things we need to do: appointments to keep, occasions to remember, notes to send, deadlines to meet, or whatever we need to devote our attention to. Such an assistant must be totally discreet about our personal affairs and on call every hour we're awake. Employing a person for this is both expense and potentially invasive. However we are already storing many of the details of our lives on our machines and can have them act as our assistant.

For example, a popular use for PDA's like the Palm Pilot is to keep one's todo list and appointment calendar. The device can notify its owner about upcoming events.

Our email presents similar problems and opportunities. We often receive letters from friends and associates that we really need to take the proper time to compose a thoughtful reply to. But we often do not have the time needed and do not want to reply in a minimal manner. To help us so we do not neglect to write a response, our software should remind

Flagged Messages

Flagged Messages --- *RonGoldman 23 Feb 2002*

. . . we keep our messages in Archives (KeepArchives) and Folders (MailFolders), and do not want important messages to become lost.

* * *

Many people get hundreds of email messages a day. Some messages they can deal with immediately, while others require more time and answering them needs to be put off till later. However it is all too easy to lose track of the pending messages amidst the clutter of a mailbox full of many, many other ones.

Some people try to solve the problem by creating a special place where they can put these important messages: a special in box. That can help, but what happens as the number of messages in that special place grows? Do you then create another really special place to store the very important messages?

Also there is more than just a message's priority that we want to indicate. Maybe it has important information, so we want it to stand out in order to find it more easily at some later date. Or within some conversational thread we would like to visually mark which messages favor or oppose an issue.

Current email clients use labels to indicate things like a message's priority, whether we have read it or not, if we have replied to it or forwarded it, if it has any attachments and so forth. The Eudora email client labels messages with one or more chili peppers to indicate that the contents might be offensive.

Therefore,

Provide a way to flag important messages so they can be easily found later. Allow messages to be labelled along multiple dimensions. Let users define new labels relevant to their life. Make sure that messages can be highlighted and sorted based on how they are labelled.

Note that a related solution is to have our computer remind us of messages we have yet to deal with---GentleReminder. That solution works best when we are being nudged about only a few messages. When there are many messages pending that we must handle in a short time frame, then the reminders would be more disruptive than helpful---and would

Conversations Threading

Conversations (threading) --- *RichardGabriel 22 Dec 2001*

... In TheContextOfOurLives every interchange has its own RhythmOfConversation. People expect a TimelyResponse, even when working in a Workgroup (GroupDiscussions)...

* * *

Electronic communication is so effective that people will sometimes carry on a discussion almost as if they were sitting together in the same room, perhaps over coffee around a table. But two things get in the way that don't in most face-to-face discussions: People will apparently stop their end of the discussion, and other discussions will be going on at the same time. These are both the result of the virtual nature of the conversations: Even though some conversations proceed nearly in real time, others are spaced out because discussants are doing other tasks while occasionally attending to the messages that make up a coherent conversation, and this causes other conversations to be interspersed with each other.

Most electronic textual communications systems receive messages in the order they are received, and some can display them in different orders or grouped by certain similarities-such as alphabetically by sender. But this can help only to some extent because the conversation threads are tangled. The same people can be involved in several conversations at the same time with overlapping people, And sometimes other apparently fixed and useful grouping indicators such as the subjects of messages will change over time as the topics of conversation change over time.

Therefore,

Make it possible for conversations to be kept separate from each other by textual electronic communications systems, especially clients. Make it possible to see clearly which messages were in response to which other and to understand easily the temporal flow of the discussion. Enable everyone to be able to see all the different views.

This can sometimes be accomplished mostly by automatic means if there were an inter-client protocol for describing the relationships between messages so that an accurate representation of the message graph can be built in each recipient client.

* * *

Keep Archives

Archives --- RonGoldman *10 Jan 2002*

. . . when we engage in GroupDiscussions or individual Conversations (ConversationsThreading) we often would like to keep a record of what was said.

* * *

Many important discussions are held electronically and it is important that the participants do not lose track of what has been said. Moreover new individuals are always joining ongoing discussions and need to understand the history of the conversation.

When we are part of a continuing online discussion we face a dilemma of what to do with the messages we receive. Do we try to keep a copy of each message? Maybe we only try to save the important ones. Or only keep them until we have answered them. It is certainly hard to delete a well written or important note that you think might be useful at some point in the future.

But even if we decide to save them, how do we organize them? We may be a member of many different groups, each of which may have multiple ongoing conversations. and if we save them how do we locate a message we only vaguely remember? We know the information we want is there but how do we find it?

When we join a new group how do we learn its history? How do we view the conversation that took place before we joined?

For mailing lists the general answer is to save all of the messages ever sent to the list in a central place and allow anyone to retrieve old messages. A great resource on the internet is the historical archive of all posts to Usenet newsgroups that includes more than 700 million messages dating back to 1981.

Therefore:

Make it easy for anyone creating a new mailing list to have messages sent to the list be stored in an archive. Provide numerous views of archived messages, some that are chronological, some in threads, by subject, by author, etc. Provide ways for people to search the archive. Allow users to annotate the messages.

* * *

Group Discussions

Group Discussions --- *RonGoldman 14 Jan 2002*

. . . NoManIsAnIsland and a major demonstration of this is the amount of time people spend talking to other people.

* * *

Much that is important in our lives takes place in the context of a larger group, be it our family, our friends, or our work. We participate in group discussions and the conversation helps to create the group.

In our lives we participate in many ongoing discussions with our family, friends and coworkers. These conversations continue even when we are physically separated from each other. In the past they took place via notes, memos and newsletters. Today they make use of email, websites and other forms of textual electronic communication.

In the real world we gather together with other people in our homes, at work, or in public places. Online a website like slashdot.org is similar to the corner pub; we can go there and converse with the regulars. Weblogs and chat rooms provide a sense of place in the online world.

A very real difficulty in the real world is often finding people interested in discussing some specific interest. Maybe this is because it is difficult to find them in a large city, or maybe because someone is located in a remote place. The online world makes it easier for people to connect around common interests through newsgroups, communities connected with websites, or even mailing lists. As long as there's a way for people to search the net they can find online group discussions on topics they care about.

Places like Yahoo! Groups let anyone create a new group, complete with a mailing list, an archive of old messages, a place to store files, a group calendar of events, a way to see the list of members, and various other features. They supply the technology, you supply the initial community.

To borrow from "A Pattern Language" [Alexander et. al.], in the online world a good group discussion helps to create an (14) Identifiable Neighborhood, which in turn helps the Internet be a (8) Mosaic of Subcultures.

Therefore,

Provide good tools to support group discussions done over computers. Support both synchronous (chat, instant messaging) and asynchronous (mailing lists, newsgroups, weblogs) ways for people to carry on discussions. Make it easy for newcomers to locate and join the discussion.

*   *   *

Be aware that SmallWorkGroupsComeAndGo and that the people in the group also come and go---EasyToUnsubscribe. Newcomers need access to previous discussion---KeepArchives and FrequentlyAskedQuestions. Provide ways to nurture the discussion, possibly creating a community---NurtureCommunity and QualityControl---Moderation,

Some discussions are private to a specific group and outsiders should not be able to join without permission. Some people may need to participate anonymously so they can be SafeToUse.

Workgroups --- *RichardGabriel 23 Dec 2001*

. . . NoManIsAnIsland, but even a well-designed and well-equipped textual electronic communications system is perhaps not adequate for all the needs of a person working in a group ...

*   *   *

People work in groups all the time: in companies and other organizations, in meeting rooms, in the workshop, and in teams. Working in groups does not require working in physical proximity, because they will use their existing textual electronic communications systems to communicate with each other.

The programming language Common Lisp was designed by a committee of about 300 people beginning in 1981 almost entirely over email using an ad hoc mailing list maintained at the Stanford Artificial Intelligence Laboratory. Since then, numerous working groups, program committees, and software development collaborations have taken place over the Internet, and a number of companies provide real-time collaboration tools for the Net. People will find a way to work together, and the advantages of the asynchronous nature of email-style of communications makes it ideal for very widely distributed (global) collaborations

Therefore,

Easy To Unsubscribe

Easy to Unsubscribe --- *RonGoldman 6 Feb 2002*

. . . in the course of GroupDiscussions the participants will come and go. Just as it must be easy for someone to join a conversation, it must also be easy for them to later leave it.

* * *

A conversation starts by people getting together and ends when we separate. This works fine when we get together in a physical location---leaving the room ends the conversation---or use a medium like a telephone---hanging up disconnects. When instead the conversation comes to us, as does email on a mailing list, then we cannot end the conversation by ourselves, instead to leave the conversation requires action by someone (or something) else.

A computer mailing list is like a subscription to a periodical. Indeed we talk of subscribing to a mailing list, a daily collection of messages is a Digest (MorningPaper), and to leave we "unsubscribe". However each mailing list has its own unique way of ending a person's subscription. When we joined the mailing list we probably did it by interacting with a web page, or maybe someone else signed us up---SmallWorkGroupsComeAndGo. When we want to leave it may be many months or years later and most folks will not remember the process they used to subscribe, and besides the subscription method often is unrelated to the process to unsubscribe.

For most mailing lists there is some email address that one can send a special message to---usually with the word "unsubscribe" as the text---that will remove one from the mailing list. A simple procedure, but one that is anything but simple to a vast number of people as witnessed by the huge number of such unsubscribe messages that get sent to the mailing list by mistake. This happens even on mailing lists where each message sent to the list includes, appended to the end, instructions on how to unsubscribe. It is somewhat comic to receive a message from someone clearly trying to unsubscribe that includes directions on what they should have done.

The typical computer-oriented response is to somehow provide better instructions---put them at the start of every message on the list in blinking, bold text! The errant user must be taught to adapt to the world of the computer. Maybe it is time to try another approach.

Therefore,

Frequently Asked Questions

FAQ's --- *RonGoldman 13 Feb 2002*

. . . all GroupDiscussions that try to NurtureCommunity face the problem of orienting new members to the group.

* * *

Because of the tremendous growth on the internet and the consequent constant influx of new people to it, for most large group discussions the newcomers almost always end up outnumbering the prior inhabitants.

Newcomers by their very definition have newly come to the topic under discussion. They may be just learning about it and they probably do not know many of the more established community members. As a result they ask questions that the old-timers have seen again and again before. They have a very real need for information, but it is often information that everyone else has no desire to see again. In fact the bulk of the messages posted to the group could easily be repetitions of previous posts.

Too little repetition and the newcomers can never advance, while too much and the old-timers will be driven away.

Since the questions from newcomers are so often the same, an early solution on many Usenet newsgroups was to collect these "frequently asked questions" (FAQ's) along with answers to them and to periodically post the FAQ to the newsgroup. This ensured that newcomers could get their basic questions answered without bogging down the general discussion. It also let old-timers reply to common questions that were still posted with the terse answer of "read the FAQ".

Web-based communities have pages with answers to FAQ's so both newcomers and old-timers can quickly find information they need. On the web FAQ's have grown to include pages with background material and links to supporting documents.

Therefore,

Provide support for the creation and distribution of FAQ's. Recognize that community members need help in writing up a FAQ and in keeping the answer up-to-date. Newsgroups and mailing lists should periodically post FAQ's or include a pointer to where they can be found in the footer of each message.

Go Away

Go Away --- *RichardGabriel 27 Nov 2001*

. . . you are adding features to complete AwarenessOfPresence in order to make the context of textual communications humane. . .

*   *   *

Sometimes our work requires deep concentration and quiet, or perhaps a planned and habitual cacophony, but not unplanned, unexpected, and unwelcome interruptions. Our trains of thought cannot stand a distraction or the nature of a textual communication would interfere with some other process we are engaging.

In some circumstances it would seem necessary to even stem the tide of knowing that a communication is desired from everyone or from a selected set of people or groups. Nevertheless, perhaps the work that requires our attention is communicating textually-composing, thinking, reflecting-possibly it is a SlowLetter we are writing. Some attack this problem with filters for individuals or by temporarily unsubscribing from a mailing list for a few hours, a few days, or a few weeks because the nature of the list is too maddening or distracting. But we do not or cannot disconnect entirely. And we do not want to lose those communications-when we are ready we want to see them.

Therefore,

Create a simple means to tell the textual communication system that its user wants part of the world trying to communicate to go away. The part could be an individual, a set of individuals, a group, or any sort of convenient way for a person to describe who is to be kept away. Create a mechanism whereby the user of the system can use it during such a period. Create a simple mechanism for seeing and interacting with those textual communications later.

*   *   *

In such a mode, a textual communication system might queue up incoming messages but not even display that they have arrived, or perhaps with a very small visual-only cue nearly out of sight. This of course would need to be able to be turned off. Filters and other programming-based mechanisms might be too hard for some people, but it might not be feasible to go as far as a simple example/negative-example learning system. This part of the system should not depend on the cooperation of a distant server.

Old News (was Go Away) --- *RonGoldman 19 Dec 2001*

. . . when we specify an InterruptionGradient we also need to indicate what to do with pending messages.

* * *

Messages have a time value. No one wants to be bothered with old news.

When we are busy, we might withdraw from the world to avoid interruptions and distractions. When we emerge later and return to the world we do not want to be overwhelmed with all that elapsed in our absence.

Everyday the newspaper, letters and magazines arrive at our door. Before we go on a trip we need to cancel delivery of the paper or we will come home to a pile of old newspapers, full of old news that we are not interested in.

Likewise in our electronic world: when we are busy or away, incoming messages pile up, demanding our attention when we return. Just as we cancel the newspaper, so we would like to be able to cancel our subscription to various mailing lists.

While we are away, our email program might unsubscribe us from whatever mailing lists we specify and then when we return, resubscribe us. Alternatively it might just throw away any messages received from those lists while we are away. Maybe it would just keep any messages received in the last day and discard older ones.

As part of our normal routine we might want our software to automatically discard old, unread messages from mailing lists rather than have them cluttering up our in boxes. But we also must be able to mark messages we want to keep. Therefore,

Recognize that messages have a time value and allow us to specify to our software what constitutes old news so it knows not to bother us with it.

* * *

Sometimes we do care about old news, so we need to be able to retrieve messages that might have been discarded---KeepArchives.

Interruption Gradient

Interruption Gradient --- *JoshuaKerievsky 04 Dec 2001*

... to be thoroughly responsive to how we will interact with our communications software---TheContextOfOurLives, AwarenessOfPresence, you are designing features to support multiple, customizable work modes. Such features will allow users to experience different levels of interaction based on their different modes of working. Given the need to fully concentrate on a message, a user will be able to fully control the degree to which they are interrupted or even tempted to be interrupted. Given a less focused state, in which interruptions are welcome, the user will be able to set their software to fully interact with them.

We love getting messages, but they are often so distracting that we can't get much done, due to constant interruptions or temptations.

Most modern email clients give users a heads-up-display that shows nearly everything that is happening with messages and connections and email folders. When new mail arrives, we're notified of that fact by gyrating icons, dialog boxes and tabulated message counts. If the email client accepts and displays web pages, we have the potential to be very much distracting by junk mail that arrives with seductive advertisements, complete with flashy graphics, stylized text and even accompanying music! How can one peacefully compose an important message amidst so much interaction?

Of course, any sophisticated email user will explain that their are many ways to customize the email program: hide this window like this, don't show that prompt by clicking this box in this options screen, etc, etc. But most users rarely explore the settings in their email's software---mostly because it involves navigating many foreign dialog boxes. And, even if they did know what to set to obtain the desired level of interaction, that interaction level would only be good for one working style and would need to be changed given a different working style. The trouble is, no one wants to constantly be clicking on settings in dialog boxes, no matter how many cool settings they let you customize.

Therefore,

Make it simple to adjust how much we're interrupted by our communications software by providing a graphical knob to control levels of interruption. Depending on where we move this knob, the software will welcome no interruptions, some interruptions or all interruptions. Let the knob settings correspond to named modes of work and let users easily customize these modes with a variety of software settings

such that simple turnings of the knob will effect numerous software settings.

The Interruption Gradient will allow users to easily enter a mode of work in which they are fully focused on one task---GoAway and will provide an easy way to not get distracted by highly stylized emails---StylizedLetter.

Given a certain interruption setting, users who may see that you are currently online, would be notified that you are currently not available for a quick chat---QuickChat. Given another setting, a chat request from a close friend would be permitted to interrupt you, while chat requests from colleagues would be recorded but not granted.

Ideally, your Interruption Gradient settings will be observed by other programs that interact with your communications software---GoodIntegrationWithOtherTools...

Interruption Gradient --- *RonGoldman 18 Dec 2001*

. . . within TheContextOfOurLives our machines need to understand what we are working on and how busy/active we are---AwarenessOfPresence. How they interact with us should be based on that information.

* * *

When really concentrating on a task, people hate to be interrupted.

Much of what we do requires us to concentrate. For many tasks we are most productive when we can devote an uninterrupted stretch of time to them. If we are constantly interrupted by others, it is hard to regain our focus and make progress. This is such a problem in the office place that many people routinely come in early or stay late, so that no one else will be around to disturb them.

People present in the same room as we are that want to talk with us can look to see if we are busy and if so wait until later. However, our colleagues and friends that choose to interact with us via telephone or email have no idea what we are doing and they cannot help but interrupt us with a ringing phone or a beeping computer.

We can always unplug our phone and disable email notifications, but we still want important messages to be able to reach us. We need to see an important email from a coworker as soon as it arrives, but other messages can wait---especially routine messages from mailing lists we subscribe to. We want our children to be able to call us, but don't want to be bothered by telemarketers.

Some email programs allow the sender to specify a priority for a message, ranging from urgent to normal to bulk mail. Unfortunately the priority is ignored when the message is received---at most it's just displayed by the client email program. Ideally a person could tell their email program that when they are busy only notify them about new mail that is marked as urgent or messages from a small list of people. New email would only interrupt them then when appropriate.

The opposite case is that urgent email arrives but we are not sitting in front of our computer. Often by the time we return, any notification that we have urgent new email has disappeared. At most we might see a special icon attached to the message in our in box. If our software is maintaining an awareness of our presence, then it can make sure to inform us when we return, and even gently remind us until we acknowledge it.

Therefore,

Make our software like an attentive butler that is aware of how busy we are and that only interrupts us for important things. Make it easy for a person to specify how busy they are and what they consider important. When we try to reach another person make it easy to specify how urgent our communique is.

* * *

One way to describe who we want to be able to reach us when we are busy is by defining an InnerCircle.

When we are very busy we might want our software to notify people trying to reach us that we are unable to answer them right now---Gone Fishin' (AwayMessages).

Sometimes our busyness lasts for days or longer and we want to minimize the messages that we receive---GoAway.

Each piece of software we use needs to respect our current interruption setting and also keep track of how intensively we are using it and share that information with other programs---GoodIntegrationWithOtherTools.

For some messages, our email software can automatically file them away for our later perusal or possibly even just delete it---MailFiltering.

Communication Door

Communication Door --- *JoshuaKerievsky 03 Jan 2002*

. . . our communications software will not fully serve our needs---PrivacyGradient, InterruptionGradient---until it knows how to distinguish messages we care about from messages we don't. This is particularly true of advertisements, which are often cast as a StylizedLetter.

* * *

Junk email distracts and disgusts us, delays us from doing our work and is delivered to some of us on a daily or hourly basis. Yet most email software provides little to no defence from it.

Junk email is on the rise. 30 messages arrived in Joe's in-box the other day. 28 of them were junk emails, 2 of them were not. The Junk emails are particularly problematic because they are stylized (StylizedLetter) and often contain repulsive or highly distracting content. So not only do people have to waste time downloading junk email, they also have to interact with it as they routinely delete it.

If we want our software to really serve our needs, we need it to protect us from such junk. Yet the protection we need must be smart enough to filter out junk, while letting new individuals reach us.

This isn't as hard to do as it may seem. First, our communications software can easily know with whom we regularly interact. With that knowledge, it can easily and automatically filter messages from known entities and unknown entities into different in-boxes. That would make for an improvement, but it would still leave us with the burden of having to check the unknown-person in-box to see if it contains junk or actual messages from new people or groups about whom we care.

What we need is some way to distinguish junk messages from genuine messages, and traditional filters aren't the answer. The solution is to automate this process as much as possible and rely on human intelligence for the rest.

Therefore,

Create a door in communication software that functions like a real door: granting access to those who have a key or know a particular kind of knock, and not granting

Intimacy Gradient

Intimacy Gradient --- *RonGoldman 4 Feb 2002*

. . . in TheContextOfOurLives we interact with many people, some we are close to while others remain strangers. In our electronic interactions with them we want to differentiate between our public and private life.

* * *

In the physical world a well designed building is arranged so that there is a gradient from a more public space (e.g. front porch) to a more private one (e.g. bedroom). This allows us to keep strangers out of the intimate parts of our lives, and to meet with our close friends and family in special places. In the electronic realm such discrimination is not always possible.

Many people carry pictures of their loved ones in their wallet or purse to show to others. Online they may post them on their personal web site, but then anyone can see them---not a comforting thought in today's world. As a result some people no longer include pictures of their children or they put them in a password protected, private section or their web site and only give out the password to their family and friends. Many companies and organizations similarly section off areas of their web presence to limit who can visit there.

A staple of online chat is the ability of two or more people to create a private chat room for themselves.

In the world of email there are mailing lists that are invitation only; not everyone can join them. Some lists (and newsgroups) are moderated to again limit they range of messages that pass over them.

To separate the private from the more public part of their lives many people have multiple email accounts. That way they can control how different people can reach them. Another common technique is to have multiple folders and have one's email client automatically filter incoming messages by who sent them---this is often very difficult to do with the available filters on current email clients. One of the reasons that spam is so offensive is that it appears in our personal electronic space in the area we associate with our friends and family.

Therefore,

Good Integration With Other Tools

Good Integration With Other Tools --- RonGoldman *19 Feb 2002*

. . . in TheContextOfOurLives we use multiple applications to achieve our goals. However to us our computer should appear as a single tool.

* * *

In the current world of computer applications it is almost as if it is our job to act as an intermediary between the various computer programs we use. Each application is often in its own world and to get data from one into another it is often left up to us to manually massage the information into a format that the second one understands---we are subservient to our own machines.

The simple task of cutting and pasting textual information from one program to another is often problematic instead of simple: font formatting is lost, paragraphs get broken into fixed length lines, extraneous white space gets inserted, and, indeed, in some programs one cannot even copy the text in the first place!

When we set our preferences for one program, it has no effect on any other applications we use. We need to tell each one separately. They were designed separately and their designers only worried about their use in isolation (or only with other applications they want us to buy from them---the all inclusive "office suite").

There are a few places where applications do seem to have been constructed with the idea that they are part of a larger world. One is that many email clients and word processors recognize URLs. They display the URL so it is highlighted, like a web browser would, and if the user clicks it then a web browser is told to display it. Some systems even let the user specify once which web browser should be used and other applications then use the selected one to display the page.

Another place where current applications recognize the existence of other programs is similar: keeping a list of "helper" applications to display files in formats the current application cannot handle. Thus an email client can launch a helper app to let a user view an attached document, or a web browser to display a downloaded file.

Therefore,

Design computer applications knowing that they need to interact with other ones. Do not

Inner Circle

Inner Circle --- *JoshuaKerievsky 12 Dec 2001*

... features that help to complete TheContextOfOurLives, including AwarenessOfPresence, PrivacyGradient and InterruptionGradient, AwayMessages can be more responsive and intelligent when there is an awareness of who is close to us or semi-close or not close at all.

* * *

Our level of familiarity with people often plays a role in how we decide to communicate with them. Yet most communication software doesn't know who is a loved one or a friend or a stranger. Consequently, the software behaves in one way regardless of who we're communicating with or who attempts to communicate with us.

Software often lacks signs of intelligence because it simply doesn't let us help make it more intelligent. We co-habitate with it but can't teach it much.

It doesn't have to be that way.

We don't expect today's software to do our thinking for us, but we can expect it to behave in ways that will enhance our lives. One way to do that is to teach the software how we would like it to behave when we are communicating with different individuals or groups.

Technically, this isn't even very hard to do. The trick is to make it simple enough for everyday users to easily control.

Therefore,

Depict levels of familiarity using a sequence of named circles---the smallest of which corresponds to our Inner Circle, the largest of which corresponds to strangers or near-strangers. Let users add, move or remove people from these circles, add new circles and change circle labels. Make the knowledge contained in the circles accessible to other features in the software, such that they may be configured to behave differently based on the identify of who we communicate with or who would communicate with us.

* * *

Intelligent Form Letters

Personalized Form Letter --- *JoshuaKerievsky 07 Mar 2002*

. . . according to ConveniencesForFastWork, our communications tool needs to help us with tasks that would take too much time for us to do manually. Producing and sending personalized form letters to a large quantity of people is such a task. We don't want these letters to be completely generic---lacking any personal warmth---and yet we don't want to take the time to personalize each letter. A Personalized Form Letter is a tool for efficiently merging the generic with the personal.

* * *

Many form letters make no personal connection with their intended readers and as a result, don't get read.

People who read form letters that have no personal connection to them often stop reading after a few sentences or words. What makes them stop? It's often a feeling that they don't want to be spending time reading a generic message that was sent to perhaps hundreds or thousands of others, and which has no direction connection to themselves.

What if the form letter mentioned the recipient's name? Would it have a better chance of being read? Probably. The simple mention of a person's name makes at least the most basic human connection. But our form letters can be more intelligent than that, for even if they mention a recipient's name they can still come across as cold and impersonal.

Our communications tools must have a better understanding of who we are sending message to in order to enable us to personalize these messages. For example, say you are sending out a party announcement to 150 people, one of whom happens to be a good friend who is also a new Dad. Wouldn't it be nice if the party form letter to the new Dad also included some text like "hey Dan, how's fatherhood treating you?" But how would your tool know to include that text? It wouldn't, but it could know that

- Dan happens to be part of your InnerCircle
- You may want to personalize form letters to people within your InnerCircle

Given that knowledge, the tool could prompt you to input personalized text to Dan and even let you decide where to include this text in the party form letter that will be sent to him. Such a capability would be a leap ahead in terms of what our email programs can do today. In fact, most of today's popular email applications don't even provide the ability to

Private Conversation

Private Conversation --- Encrypted --- *JoshuaKerievsky 09 Jan 2002*

. . . How could someone feel comfortable sending someone else an electronic message when that message contained sensitive information or data?

* * *

Everyday email and chat programs fail to provide us with an easy way to conduct a private conversation. The encryption software is available, but it isn't integrated with our software and it isn't easy to find and integrate it with our software.

Yeow in China was going to take a class offered by Joe in San Francisco. Yeow needed to pay for the class and Joe's web site didn't support secure payments. Yeow didn't want to make an expensive phone call to Joe just to give his credit card number (CheaperToUse), and yet he also didn't want to send his credit card information via a plain-text message, since such a message could be intercepted by some shady character. So what did Yeow do? He sent an email with the first half of his credit card number on Monday, and another email with the 2nd half on Wednesday, figuring that it would be unlikely for a hacker to put two email messages together to obtain his credit card number. Yeow's technique worked, but it is a poor work-around for software that doesn't meet our needs.

It is a fact of life that we sometimes need to speak privately with others. Our electronic communication software must respond to this need.

But what about those who are bent on evil? Should our communication software enable evildoers to speak privately with each other? Some would answer that question with another question, for example: "Does a government have a right to protect its citizens?" Quickly, this question can dissolve into a heated debate about rights and freedoms, national and international. We won't go there.

Instead, we'll ask this: what kind of privacy does a common user need? A common user needs to occasionally send private messages that aren't impenetrable to spying eyes, but which are difficult enough to crack that most folks wouldn't bother.

Therefore,

Make it easy to send and receive private messages by fully integrating sophisticated, but crackable, encryption software into our communication software. Make this

Safe To Use

Safe --- *RonGoldman 7 Dec 2001*

. . . when a person chooses to share their life (TheContextOfOurLives) with others they have every right to demand that the technology involved will keep them safe from harm.

* * *

Everyone has a fundamental right to safety.

As we trust more and more of our lives to our technology it is vitally important that our technology be designed to keep our online information safe. We should not have to be afraid that exchanging documents or email with a friend will cause harm to our electronic self.

We also should not have to fear that someone else can commandeer our online identity and besmirch our reputation by impersonating us.

Current computer software is often like a house with the front door wide open; it seems designed more to empower malicious ne'er-do-wells then to keep us safe. We are continually reading in the news about some new computer virus or worm wreaking havoc in the online world. The world's largest software company seems to supply the most unsafe software.

While it is probably impossible to create a software application that is totally safe, it is straightforward to do vastly better than is currently done. The basics of security, such as preventing buffer overruns and setting defaults to disable dangerous features, are well known and should always be used in practice. The vast majority of people will never want to have a macro in a text file or spreadsheet that automatically sends email to everyone in their address book. For those few that do, their software can ask them for permission.

Another aspect of safety is from ourselves: our software should warn us before we do anything that is potentially damaging and unrecoverable such as deleting all of our files.

Therefore,

Build software that by default will protect itself (and hence us) from any malicious attacks that might be launched against it.

One Person Many Machines

One Person, Many Machines --- *JoshuaKerievsky 03 Dec 2001*

. . . Our communications software must be responsive to TheContextOfOurLives. One such context is the ever changing landscape of machines we work on. Good communications software will give us universal access to our textual electronic communications, freeing us from dependencies on particular machines.

* * *

If you routinely communicate with people from different machines, it's likely that your communications will be scattered around on various hard-drives, unreachable when you aren't near a particular machine.

The vast majority of us who do communicate with people from different machines are compelled to invent ways to cope with our scattered messages:

- Linda is going on a trip with her laptop and she needs access to some important messages she's exchanged with her client. She doesn't know how to easily copy message from her desktop to her laptop, so she spends the time to print out each message and stows the paper documents in her luggage.

- While on her business trip, Linda uses her laptop to communicate with clients, loved ones and friends. When she returns to her office, she goes back to using her desktop. One day she needs to recall what she had quoted to a her client. She can't find the quote in any email on her desktop machine, so she goes digging around for it on her laptop.

- On another trip, Linda's screen on her laptop never turns on, making her machine useless. When she returns from her trip, she finds out that the cost to repair the machine is close to the cost to buy a new one. She decides that it would be better to buy a new one, even though she'll loose many important messages and documents on her laptop.

Stories like this are commonplace when people interact with communications software from many machines. Of course, some figure it all out---they come up with just the right way to use their machines and software to make all their messages accessible to them. But these folks are in a tiny majority of folks who either learned how to do this from a someone else, or had the technical expertise to make it work.

Therefore

Make textual electronic communications accessible from our various machines, but not dependent on those machines for the storage of our messages.

* * *

Machines can often fail us in remote locations, when access to our messages can be important. At such times, communications software systems can cooperate with non-software systems---like faxes, phones or even postal mail---to give us access to our messages.

Good communications software will know how to interact with us as we move from machine to machine---UniversalAccess, AwarenessOfPresence, TimeSensitiveAndLocationSensitive.

Electronic Nomads (was One Person, Many Machines) --- *RonGoldman 6 Dec 2001*

. . . the pattern TheContextOfOurLives provides for programs to act appropriately in different contexts. This context goes beyond the here and now, as tasks we are engaged in extend over days, weeks and even years. As we move about it also extends to the many different computers and electronic devices that we encounter and use.

* * *

To fully function in the world we rely on external artifacts.

As we work and play we do not attempt to remember everything in our heads. Instead we depend upon address books, calendars, notes, drafts of work in progress, filed copies of our correspondence, books, and other external memory aids. Likewise we have our favorite tools, be it a fine pen or a power saw.

Depending on external technology creates a problem when we move about in the world. We either have to take our artifacts with us or hope to make use of those available wherever we are going.

The same is true in the electronic realm: when we move from one computer to another we need to bring with us copies of our letters and notes, along with the programs we have learned to use.

When we start interacting with a new computer, it has no knowledge of the context of our lives. We must start from scratch, copying our world to it. We must install the software we wish to use, copy the information of who we are and what we are working on, and re-establish our preferred working environment.

If we only seldom needed to shift to a new machine then the problem, while an inconvenience, would be bearable. However people often use many different machines in the course of a single day. They may shift from a computer at work to their portable laptop or to a smaller PDA. They may sit down at a coworker or friend's machine. They move from home to work. Over time old machines break or become obsolete and are replaced with newer designs.

This nomadic lifestyle leaves us with information about our lives scattered amongst all of the machines we use. A letter we just received is on our home machine, our latest notes concerning a recent meeting on our laptop, but now we are at our work machine and need access to both.

We need to enable our machines to talk to each other in order to always allow us access to our work. This inter-machine communication should occur automatically rather than force us to attend to the details manually. There might be some central---possibly web-based---place that contains our online lives, or just a way for one machine to inform or synchronize with another. The idea of turning off my home computer when I'm not using it may no longer make sense, if when I am at work, my work computer needs to retrieve information from my home one---when I use any computer, I may indirectly need to use my home machine.

Similarly when we approach a strange machine, it should be able to "learn" our preferred work practices and be able to access our information. It needs to be able to converse with some other machine that already knows us that can tell it about us.

Therefore,

Build into every program a way for it to update and be updated about work we do on other computers. When we change our online world on one machine that change should be propagated to any other computer we use.

* * *

Making it easy for our machines to share information about us must be balanced with concerns about our privacy---PrivacyGradient---and our safety---SafeToUse. Being able to use the same tools wherever we are ties into EditingEase.

In the longer term as our older machines cease to serve us, we face the loss of the tools (software) we know and (possibly) love. Shiny new replacements are unlikely to run the old applications, and we will be lucky if the new versions can make sense of the old formats that our lives are stored in---RosettaStone.

One Person, Many Machines --- *RichardGabriel 7 Jan 2002*

. . . Machines form part of TheContextOfOurLives as important tools, more important in some ways because we communicate so much with them, and the RhythmOfConversation doesn't heed how we switch from one machine to another, just as it doesn't heed one pen replaced by another . . .

* * *

Many people work from many computers, even during the same day. We like to work particular ways, and like dogs pressing down straw to make a nest we spend time customizing our machines for our best and most comfortable use. What data and information we keep on each machine is part of our context, but most of the time we don't associate that context with the machine.

When I spend a week away from home, using my laptop to communicate, the people I communicate with don't care or even know something is different from when I am home, or even that I am away and not at home except that my rhythm might be off or the hours I communicate are different by time zones, travel, and meetings.

A machine is a machine, but I and my context is what matters.

Therefore,

Arrange the textual electronic communications system so that when a person moves from one computer to another, all the personal preferences for how to communicate are moved from the one computer to the other. Further, make it so that all the archives, saved messages, partially composed messages, address books, presence settings, filters, and everything else associated with a person's use of the communications system is automatically moved from the one computer to the other.

* * *

Some systems call this synchronization, but it needs to go beyond data files to include personalizations and executable code that implements preferences. This will undoubtedly

Privacy Gradient

PrivacyGradient --- *RonGoldman 14 Dec 2001*

. . . within TheContextOfOurLives our machines need to understand who we are willing to interact with. As we move about---Electronic Nomads (OnePersonManyMachines) and AwarenessOfPresence---we want our programs to respect our privacy.

* * *

People have a need for privacy. We do not want everyone to always be able to peer at us. In our physical architecture our rooms have doors that we can close, and windows have shades that we can draw. The best buildings have an Intimacy Gradient (Alexander, APL) to separate the public from the private parts of our lives.

When you are online you similarly might want to expose more or less of what you are doing to other people. For instance if we are on vacation---Gone Fishin' (AwayMessages)---we may want our family and close friends to know our schedule and how to get in touch with us, while we do not want that information revealed to strangers or casual acquaintances. We may want to restrict access to our online calendar to just the members of our project team. Our machines should be able to distinguish who is asking, and provide appropriate answers.

Software for many business applications currently uses authentication to confirm someone's identity and authorization to control what information they can see and modify. We need to extend this ability to our personal online lives.

Therefore,

Let people have control of who can see any of their personal information that is available online. All programs need to respect the access policy that a person chooses.

Contrast this pattern with InterruptionGradient that allows us to control which incoming messages and requests will be brought to our attention. PrivacyGradient gives us control over what information about us is broadcast out to the rest of the world.

* * *

People might want to specify policies for groups of people by defining an InnerCircle.

Return Receipt

Return Receipt (was Notify User if Mail has been Read/Scanned) --- *RonGoldman 10 Jan 2002*

. . . The RhythmOfConversation is sometimes slow and we would like some way to make for a more ExpedientResolution.

*   *   *

When we send an important message we want to know that it reaches the intended recipient.

If we post an important letter in the mail we may pay extra to receive a return receipt. It provides us proof of delivery. The return receipt shows who signed for the item and the date that it was delivered. The return receipt may also inform us of an updated delivery address if the letter needed to be forwarded. We don't know that anyone ever opened and read our letter, but we do know that it arrived into their hands.

With email we can do better. We can have the receiving machine notify us when the email arrives, and we can further be notified when the mail is finally read. Or at least we can request to be notified when the mail is read-some people value their privacy and will not want their computers reporting to others on their actions.

Therefore,

For any asynchronous communication system build in a way to allow the sender to be notified that their message was successfully delivered and when it is actually read by the person who has received it. Do so in a way that allows the receiver to preserve their privacy, if they so choose.

*   *   *

Make sure that the receiver remains Safe (SafeToUse), the default behavior for an email client should be to ask for permission before sending a return receipt notification.

Consider using FlaggedMessages as a way to indicate to the sender that their mail has been received/read.

People Talk To People Machines Talk To Machines

People Talk to People, Machines Talk to Machines --- *RonGoldman 12 Dec 2001*

. . . we are concerned with TheContextOfOurLives, and so then must our machines. But our machines also have their own contexts that they must deal with. When our machines have an AwarenessOfPresence they may need to share that information with other software. When we've Gone Fishin' (AwayMessages), our machine may need to let other machines know that we are away.

* * *

The inner workings of computers are filled with details that we do not want to be bothered with when we are using them.

We use our technology as a tool to communicate with other people and as a tool to do our work. Good tools are transparent: They don't try to strike up a conversation with us, but quietly do our bidding. Only if problems arise do they need to inform us---and then in terms of the human world, not in machine terms. So never as an error code like "message had fatal error: 550", but rather in the context of what we were trying to do, like "no user named 'joe' was found at 'fabitz.com'."

Our current systems don't respect this distinction and so assume any mail we receive needs to be dealt with by us directly. With dumb software that is the only safe choice, but by marking all messages with some type of meta information, our software can distinguish between messages sent by an individual, by a mailing list (and which one), and by a bulk mailing (spam). In that way our software can begin to act more intelligently. For example, our email program would know not to send our Gone Fishin' AwayMessages in reply to a message sent to a mailing list we are on.

With the proper meta information attached to a message it becomes much easier for us to have our email filtered and sorted as we would like. Putting all the messages from one mailing list in a special folder is straightforward. When our machine receives a ReturnReceipt, it can then just flag the stored copy of the outgoing message as having been received. It is easy to keep track of the replies to each message and store them as threaded conversations (ConversationsThreading).

Therefore,

Don't have programs throw away the human context when they interact with another program. Design programs to share information with other programs about the context that they are acting in.

* * *

Sometimes our machines need to talk to us---GentleReminder---while other times we want them to talk to other people in our stead---WhenOurMachinesActForUs. For software to talk to other software requires GoodIntegrationWithOtherTools.

People Talk to People & Machines Talk to Machines --- *RichardGabriel 27 Dec 2001*

. . . when designing software that interacts with people . . .

* * *

There are many worlds of conversation that intersect only on rare occasions: Dogs growling and people taking heed of the danger, storm clouds climbing above the trees and people taking in their laundry. Computers don't speak the language of people.

Some interactions normally take place within their unique domains: Birds interact with birds, people with people. Computers shouldn't talk to people unless asked. Some principles transcend any application domain. When a computer talks to a computer, only programmers and sys admins might want in on the conversation.

Therefore,

Never design software to bother a person when a computer talks to a computer unless the person wants to listen in. Computer talk spilling into the domain of human consciousness without an explicit design requirement for it is a design error of the first magnitude.

* * *

AwayMessages are the result of an explicit requirement for people to be able to ask their computers to intercede on their behalf to inform other people that they are away. However, when a mailing list server sends a message to a mailer, that mailer in sending back an ?AwayMessage is talking to another computer, and there is no explicit requirement that a mailing list---sometimes vast in its reach---should receive such an inconsequential communication.

The Context Of Our Lives

The Context of Our Lives --- *RonGoldman 29 Nov 2001*

. . . you wish to create a technology that supports each person's unique, individual way of working and living as they shift from one task to another, move from location to location, and interact with various other people.

* * *

Our lives are varied and complex---actions do not occur in a vacuum, but in the context of other actions. At any moment we are focused on the task we are working on. Other tasks that we may have pending are temporarily forgotten.

Most software is focused only on the task it performs. It does not include any awareness of activities the user does outside of it. As a result programs are constantly acting inappropriately. Ideally when we pick up the phone, if the radio is playing it should lower its volume. If we are intensely concentrating on a task, perhaps the phone should not even ring---unless the call is very important.

The ideal butler is highly attentive to your needs and also unobtrusive in carrying out his activities. Our technology is totally ignorant of us and what we are doing; all it "sees" are its needs and actions.

Therefore,

Build programs that recognize the different contexts and act appropriately. They must include information that goes beyond the narrow tasks they perform, so that those tasks can be fitted into the larger context of our lives.

* * *

Our machines need to cope with many people using the same machine---MultiplePersonalities---and one person using many machines---OnePersonManyMachines. They must respond differently if we are present or absent---AwarenessOfPresence---and depending on when and where we are using them---TimeSensitiveAndLocationSensitive. They need to understand who we are willing to interact with---PrivacyGradient and that while sometimes we do not want to be interrupted, other times we welcome a break in what we are doing---InterruptionGradient.

They also need to understand the difference between messages from other people as opposed to another machine acting for another person---PeopleTalkToPeopleMachinesTalkToMachines.

One way to convey the larger context is to use MetaInformation.

The Context of Our Lives --- *RichardGabriel 8 Dec 2001*

. . . we need to communicate (NoManIsAnIsland), we need to use tools sometimes to do that (the telephone, letters, computers), but no one wants tools to dictate how to live . . .

* * *

Our lives are varied and complex. Actions don't occur in a vacuum but in the context of our lives, other actions, situations, and moods. Our lives change-sometimes dramatically and suddenly. But tools just sit there. They were designed long ago and far away. And their designers assumed you would be using them in a particular context.

But software isn't like an ordinary tool. It can be aware of your context or situation-especially if you you are willing to tell it. If only its designers would think of this. The failure of many software tools is based on assuming a fixed context.

For example, some folks like to automatically notify others that they are away-on vacation, on a business trip, etc. So they set up their emailers to auto-email an away message. But they're also subscribed to email lists. Uh Oh. Now everyone on the list gets a message about someone being away-every time they send a note. How embarrassing.

Therefore,

Try to build the textual electronic communications system so that it recognizes different contexts and acts appropriately. Our software tools are like our surrogates. How embarrassing when they act funny.

* * *

AwayMessages are an important first step in recognizing context. AwarenessOfPresence helps the textual electronic communications system react to incoming messages based on whether we are there or not. Sometimes we'd rather be left alone (GoAway, PrivacyGradient). We also use many machines and want them to act as one

Meta Information

Meta Information --- *RonGoldman 21 Feb 2002*

. . . all communications between people takes place in TheContextOfOurLives. The contextual information is as much a part of our communication as the messages themselves.

* * *

When we pick up an old letter we received years ago it may cause us to remember all that was happening back then: the person that sent it, how we felt, what our concerns were then, who we were even. Other times we are not so lucky and we are left wondering: who was it from, whatever was it to us....

How many times have you looked at an old photograph and wondered who were the people in it. At the time it was taken we may have thought we'd always remember, but we don't always do so. How wonderful then if we turn it over and see we left a note to our future self giving names to the faces and a description of the event recorded.

It's not just as an aid to our memories that we want information about our communications. When we get a phone call how convenient to have caller id let us know who is calling before we are committed to talking to them.

In our email it is a help when our email client application shows us whether we've read a message yet or if we've replied or forwarded it. Seeing who a message is from and what the subject is helps us to prioritize the order we read our mail. Being able to sort our messages by date or sender or subject aids us in organizing our correspondence and finding old messages---studies have even shown that people make more use of sort than they do of search.

Other useful "meta information" about our email is to connect a message sent in reply to a previous email. How much better would it be for us if our email clients let us easily view the thread of messages that make up a conversation---ConversationsThreading. Newsgroup readers often use the information of what postings are replies to others to allow them to display related messages in the order they were posted. That makes it much easier for us to later follow the course of the online discussion.

Keeping contextual information about a message is useful for both us and our machines. It lets us better understand and navigate our messages. It also enables our machines to act

Quality Control

Quality Control---Moderation --- *RonGoldman 14 Feb 2002*

. . . in GroupDiscussions involving many participants it takes effort to keep the discussion on track and focused.

* * *

Even in a small group where people share a common agenda, keeping a discussion on topic is difficult. In a large group, particularly one composed of strangers where everyone has different interests and their own idea of what is relevant, it can become impossible.

An online group may have thousands of members and a each new topic might result in thousands of replies, many of which may not be very interesting---in fact most will be pretty awful---but some will truly be gems. With such a large signal-to-noise ratio only the most dedicated readers will ever see the insightful or wonderful messages, the sheer torrent of garbage will drive away everyone else.

Online discussions have additional problems as participants do not get any nonverbal cues from the other group members and the asynchronous nature of newsgroups and mailing lists further fractures the conversation. Also online social norms are less in effect and normally polite individuals are apt to flame or interrupt. The online world also suffers from electronic junk mail, known as "spam."

In real life when people hold a meeting they often have one person act as a moderator to facilitate it. The moderator decides who gets to speak, and makes sure that each speaker stays on the current topic and does not digress. The moderator can cut off anyone who starts to go off topic.

For an online discussion having a moderator can also help. Early on in the history of Usenet some newsgroups chose to have a moderator. In such groups all messages first go to the moderator who must approve them before they are posted for all to see. Those messages deemed inappropriate, for whatever reason, will never be seen by the rest of the group.

Most moderators just act as a gate, filtering out the "noise". Some take on the role of editor, taking all the messages from one day, grouping them by topic, eliminating redundant sections, and posting the resulting digest of approved messages. Needless to say being a moderator can be very time consuming, especially for an active discussion

Nurture Community

Nurture Community (was Host and Introductions and Culture) --- *RonGoldman 11 Feb 2002*

. . . GroupDiscussions involve more than just the exchange of messages, they are also about creating community.

* * *

"A community is a group of people with a shared interest, purpose, or goal, who get to know each other better over time." [Amy Jo Kim, "Community Building on the Web"] But getting to know a group of people when all of your interactions are via computer can be difficult.

For a lot of people moving to a new town where you don't know anyone can be a lonely experience. Even attending a party with people you do not know can be difficult; there are people to talk to, but you don't know them or their interests. At a good party there is a host who will introduce you to different folks and maybe even point out common interests you share.

The online world also benefits from people acting as hosts that greet newcomers, help orient them, introduce them to other members of the community, and generally keep things running smoothly. The host, and other community elders, help establish the tone for the community, providing an example of how people should act.

The underlying technology can also help to orient newcomers to an ongoing community. For example, on IRC chat there is the "whois" command to find out a person's name and email address. Some chat systems let people specify other information to be included with their "whois" information such as the URL for their home web page, a list of their interests, or whatever they want other people to know about them. Some web-based discussion groups, such as Yahoo! Groups, let people see a list of all of the group's members that includes links to their personal profiles pages.

Some mailing lists have a culture that encourages people to introduce themselves the first time they post to the list. Those messages are often saved, so newcomers can always search the archives to find out more about other, more visible members.

Each community has its own unique culture with its own phrases, in jokes, rituals, and customs. Some of this can be recorded in files or web pages that newcomers can then read

Rosetta Stone

Rosetta Stone --- *RonGoldman 6 Dec 2001*

. . . being Electronic Nomads (OnePersonManyMachines), we move from machine to machine and from one software application to another. In each environment we need to be able to access the information we have stored online. We also need to be able to share information with others, including basic messages, StylizedLetter, drafts of papers, and spreadsheets.

* * *

When we write down information somewhere, we expect to be able to read it at a later date.

Fifty to ninety percent of the world's languages are predicted to disappear in the next century, many with little or no significant documentation. (http://www.rosettaproject.org:8080/live/concept) Whatever texts are written in those languages will become unreadable when no one is left alive that can speak them.

The situation for information stored online is equally grim, only the timescale is years rather than centuries. Most software uses proprietary formats the details of which are not made public. In fact recent laws such as the Digital Millennium Copyright Act (DMCA) may even make it illegal to reverse engineer a proprietary format. It is common for new versions of software to not be compatible with previous versions, forcing people using the old version to upgrade if they wish to read files sent them by their colleagues who are using the new version.

All in all if different software (email clients, editors, etc) is being used it is quite difficult for someone to exchange information with other people, or even with themself when they are using another machine.

In the dawn of the computer/internet age the limitations of our machines may have made it necessary to conserve each byte of storage, but that is no longer true. We need to store information in formats that contain a description of what they contain and how it should be displayed. For example, XML-like technology is designed to provide MetaInformation that describes documents and makes them universally accessible.

Programs can then interpret as much of this information on how to display the text as they are capable. Meta information they do not understand can be ignored.

Sharing Information

Sharing Information --- *RonGoldman 16 Jan 2002*

. . . when we talk with others we often need to exchange more than just a simple message.

* * *

There's more to our lives than what we talk and write to each other.

When we communicate with each other we often need to refer to existing documents, photographs, computer files, voice messages, links, etc. One way we do this is by sharing them with the recipient of our message. Most modern email clients allow a message to include any computer file as an attachment. Assuming the recipient has the appropriate program, they can then view the document.

According to recent research email is now the main means of document exchange. ["E-mail as Habitat: An Exploration of Embedded Personal Information Management" by Nicolas Ducheneaut and Victoria Bellotti, in interactions, Vol. VIII.5, September + October 2001]

Sometimes only a single line of text is required to provide a pointer to the actual information, as with a URL for a web page. Other times we need to share many large files that make up a document.

The information we are sharing may be limited to the message, such as an accompanying picture from our vacation. Or it may be information that the receiver will need to file away to work on later, such as a paper being written in collaboration with the sender.

Therefore,

Make it easy for people to include non-textual information with their messages. Let them integrate it as part of the message or keep it separate as an attachment. Make it easy for the receiver of a message with attachments to view and store them away.

* * *

The information we exchange is often in formats that our email client will not understand so it is important that the appropriate application to display the information be able to work with our email client---GoodIntegrationWithOtherTools. To be sure that the

Time Sensitive And Location Sensitive

Time Sensitive and Location Sensitive --- *RonGoldman 10 Dec 2001*

. . . as we move through the day and about in the world TheContextOfOurLives changes and our software must reflect this. Being Electronic Nomads (OnePersonManyMachines) our technology needs to take account of our location.

* * *

The world is rich and varied, changing as one moves from place to place and also as one moves through the day. What we choose to do reflects where we are, both our location and the time of day.

We change our behavior so it suits where we are; our software does not. We adapt to whatever resources may be present; our software does not.

For example, if I am talking to someone over the telephone about a paper they are interested in, I will likely summarize it or just give them a reference so they can follow up on it later. If my computer is connecting to my mail server over a low speed link, say a 28.8k modem, and I have new mail with a large file attached to it, then shouldn't my email client be aware that it will take a long time to download the attachment? Shouldn't it tell me who the mail is from, how long it will take to download, and then ask me what to do? And while it's waiting for my reply, shouldn't it proceed to retrieve my other pending messages?

Current email applications assume that the context is always fixed. They assume we set our preferences once and for all and then everything is fine. This was reasonable when our machines were fixed in place and our options were limited. Now that we have portable machines that connect to the internet in a variety of ways, the software needs to be more flexible.

By studying how a person's needs change in different situations, software can be designed to act to meet those needs. At the least an email client needs to act differently depending on whether it is connected to a mail server or not (online vs. offline), on what speed connection is available (slow vs. fast), and on how much information it needs to send or receive. For each location we frequent, the email program needs to know what mail servers are accessible and how to reach them.

As we carry our laptop from place to place, it should identify when it can connect to our mail servers and automatically do so, updating our in box with any new messages. When we arrive home or at work it should be able to synchronize with our desktop machines.

If we leave our computer running overnight, then our email program should know that we are likely sleeping and so it should not use sound to announce new email---unless it detects that we are currently using the computer (e.g. typing on the keyboard or moving the mouse).

It should be easy for a person to enter their schedule into their computer. Then they can expect it to quietly power on just before they get up in the morning and have their email waiting for them. Likewise their machine can know to power down automatically if left on at night. Many computer stores set up the display computers to all start up just before the store is scheduled to open and to power down just after it closes.

Therefore,

Build into programs multiple contexts that identify and take into account our location, the time and the local resources. Let the user specify what behavior is appropriate for each context.

* * *

Sometimes we are away from our machines for periods of time and they need to act in our behalf---AwayMessages.

Time Sensitive and Location Sensitive --- *RichardGabriel 9 Jan 2002*

. . . The RhythmOfConversation goes on and rolls over our disappearances and peripatetic natures to catch people in differently equipped situations. We have started to think about AwarenessOfPresence and turn our attention to the awareness our machines have of where they are . . .

* * *

The relevance and importance of a message depends on where you are when it arrives and how it fits into the conversations you are having. A message from someone you've never or rarely got a message from that contains a lengthy attachment is sometimes not welcome if you have just dialed into a 28.8kbaud modem just outside Tulsa while driving cross country to an old timey vintage clothing swap. But if you get a message with a lengthy attachment from someone you've been corresponding with all the way across,

Multiple Personalities

Multiple Personalities --- *RonGoldman 30 Nov 2001*

. . . you wish to build a program that reflects the richness of our lives so it will act appropriately in different contexts---TheContextOfOurLives.

* * *

We present different sides of ourself to different people. Our identity is plural, not singular.

For example, people customarily have several email accounts: one email address for work, another for a personal account, one given out to online merchants, another to a special interest group. Most email programs ignore this fact and make it difficult to switch from one address to another.

Each identity may also involve preferences on how various programs should be configured. This is particularly true when multiple individuals use the same computer. Each person will have their own preferences.

Therefore,

In any program that involves our online identity include the capability for easily switching between various identities.

Multiple Personalities --- *RichardGabriel 11 Feb 2002*

. . . In TheContextOfOurLives people shift identities. Building a textual electronic communications system, don't be fooled by identity . . .

* * *

When it comes to electronic communications, many people have multiple identities. A employer may give you one electronic identification while you might have several of your own. I, for example, am rpg@eng.sun.com, Richard.Gabriel@sun.com, rpg@dreamsongs.com, and nickieben@earthlink.net. From my computer I am able to connect to all these places and read email. I want, however, to use one program to do that, and without having to go through a lot of steps to switch personalities.

When a person acts in a role, we call that a "personality," because we are not pretending to be someone different, we are someone different because we are acting in a different context. How I act, what I do, and what I say at work is different from those same actions when I'm at a poetry workshop.

The engineering details of email and messaging can get in the way. Some places use IMAP, others POP. According to Christopher Alexander, engineering should never dictate social spaces: "a first principle of construction: on no account allow the engineering to dictate the building's form." Although multiple steps might be needed to effect the change from one personality to another, I want to do only one.

On the other hand, some personalities may not be able to be "invoked" in all circumstances. When I am at Sun, I cannot fully be rpg@dreamsongs.com or nickieben@earthlink.net, because I cannot access my email from within Sun's firewall. And in some cases, rpg@dreamsongs.com may not be allowed to know what Richard. Gabriel@sun.com knows, or at least it cannot appear that way. However, in some cases a person in a single location can receive messages from many if not all personalities. For example, while at home I can easily receive email sent to rpg@dreamsongs.com, and I can easily receive email as rpg@eng.sun.com.

However, to be able to turn from receiving messages from one of these to receiving from the other requires me to switch my TCP/IP settings, invoke a VPN client, log in manually, and switch my mailer settings. VPN is secure connection, so no one allows the process to be entirely automatic. However, allowing a personal item to stand in for the person is as secure as one can ever get, so it should be allowed.

Regardless of which personality I am, I am always myself, and sometimes messages received or sent as one personality are relevant to another. I should be able to observe merged or separate message tracks.

Some systems have "location managers" that solve some issues attendant to this problem, such as changing network connection settings when you move with your laptop, which might be part of a personality switch-for example, when I go to my Sun office, I need to switch not only who I am but how I'm connected.

Identifying personalities and connections seems to be difficult for software. In Eudora, I can't completely switch to being a totally distinct personality, but somehow I must always remain a variant of myself. For example, while at Sun, I could become rpg@dreamsongs. com were I able to receive email through POP and send it through IMAP, but Eudora doesn't contemplate this. With the Apple location manager it is never clear how I change

Richard Gabriel

website: http://www.dreamsongs.com

Richard P. Gabriel received a PhD in Computer Science from Stanford University in 1981, and an MFA in Poetry from Warren Wilson College in 1998. He has been a researcher at Stanford University, company president and Chief Technical Officer at Lucid, Inc., vice president of Development at ?ParcPlace-Digtalk, a management consultant for several startups and Sun Microsystems, and Consulting Professor of Computer Science at Stanford University.

He helped design and implement a variety of dialects of Lisp. He is author of three books ("Performance and Evaluation of Lisp Systems," MIT Press, "Patterns of Software: Tales from the Software Community," Oxford University Press, and "Writers' Workshops and the Work of Making Things," Addison-Wesley Press), with two forthcoming in 2003-2004 ("Innovation Happens Elsewhere: How and Why a Company Should Participate in Open Source," (Morgan Kaufman), and "Leaf of my Puzzled Desire," (Dreamsongs Press)). He has published more than 100 scientific, technical, and semi-popular papers, articles, and essays on computing.

He currently is a Distinguished Engineer and chief scientist of a small laboratory at Sun Microsystems, researching the architecture, design, and implementation of extraordinarily large systems as well as development techniques for building them. He is Sun's open source expert, advising the company on community-based strategies. He is also President of the Hillside Group, a nonprofit that nurtures the software patterns community by holding conferences, publishing books, and awarding scholarships.

He is the lead guitarist in a working rock 'n' roll band and a poet.

Notes For Ones Self

Notes for One's Self --- *RonGoldman 18 Jan 2002*

. . . in TheContextOfOurLives we need to manage our personal information.

* * *

We cannot remember everything that we need to.

In addition to communicating with other people, many people also use email to send notes to themselves. They jot down things they need to do, reminders and whatever else they want their future self to attend to. A study by Xerox PARC researchers found that people tend to live in their email program, using it for all sorts of personal information management such as their appointments, todo lists, etc. ["E-mail as Habitat: An Exploration of Embedded Personal Information Management" by Nicolas Ducheneaut and Victoria Bellotti, in interactions, Vol. VIII.5, September + October 2001]

Email address books get used to hold other contact information such as phone numbers, street addresses, or general comments.

Appointments and todos are often kept in one's "In" box as a constant reminder, though they may be buried by other recent messages, or they may be kept in special folders.

Therefore,

Recognize that people will use any email client for more than just exchanging email with other people. Make it easy for people to create notes for themselves and to edit them later. Build in ways for people to include non-email information in address books.

* * *

Some of this personal information can be captured by allowing people to flag and annotate messages (FlaggedMessages & AnnotatedMessages).

Our machines can help to remind us---GentleReminder.

Instead of duplicating the features of calendaring software and other personal information management applications make sure that the email client connects seamlessly with those

Whos Who

These people founded this work and continue to contribute to this site.

- RonGoldman
- RichardGabriel
- JoshuaKerievsky

These people support this project in one small way or another.

- WardCunningham

Edit

Last edited December 25, 2002
Return to WelcomeVisitors

System Administrator

The administrator(s) of this site can help if you forget your password. Please note their names and phone numbers where you can find them should you be unable to reach this page again.

- WardCunningham

(We have chosen to not password protect this site. We've made this choice for the convenience of the authors. We hope all visitors will edit pages here whith the same care they have shown this work.)

Special links for use by administorators:

- http:pass.cgi?JohnDoe -- to change any password
- http:user.cgi -- to add or remove users

Edit

Last edited December 25, 2002
Return to WelcomeVisitors

Ward Cunningham

7830 SW 40th Ave
Portland, Oregon 97219
503-245-5633
ward@c2.com
www.advogato.com

My work

- Cunningham & Cunningham, Inc., my firm.
- Portland Pattern Repository, my contribution to the web.
- WikiWikiWeb, the technology behind PPR.

My family

- Karen, my wife.
- Patrick, my first son.
- Christopher, my last.

Action at a distance

- Morse Code Teaching Machine and other ham related software. *(download)*

Rhythm Of Conversation

Rhythm of Conversation --- *RichardGabriel* *Nov 2001*

. . . you are laying out the design principles for a textual electronic communication system and desire to make the most congenial system you can for the type of use it will be put to . . .

* * *

Different conversations have different rhythms. Sometimes the exchange is slow, with time to think between each reply and a desire to carefully consider each letter and its appropriate reply. Other times the exchange is fast and furious, with ideas sent as fast as possible, perhaps for the purpose of getting quick action, deciding a relatively unimportant matter, or to have a fun, fast exchange.

It is easy to accidentally discourage the natural rhythm of the conversation by offering or emphasizing inappropriate behavior and interfaces. For example, in a communications system that assists fastpaced conversations, it is useful to have aggressive auto-completion, spell-checking and spelling correction, a palette of emoticons or other special abbreviations---in a fast exchange, these can both speed up and enhance the quality of the message. In a slow-paced conversation, the writer is interested in reflection and in creating a good-looking, well written letter. In such a situation, having indicators popping up displaying mistakes or auto-completing are distracting and therefore discourage the natural rhythm and frustrate the writer. In fact, in such cases tools for checking things like spelling should wait in the wings until the writer is ready for them rather than intruding on the process.

Another example is the use of pen and paper. In such communications, the physicality of interaction forms part of the rhythm of conversation---the act of moving a pen on a scratchy piece of paper reinforces the intimate and delicate human-to-human nature of the conversation. The physical motion requires more time and encourages thought before writing, as does the difficulty and ugliness of trying to erase ink. In this case, the deliberate difficulty of using the mechanism of interaction encourages a welcomed rhythm.

Some conversations are specialized enough---such as calendaring---that non-local conversations about when and where to meet are best mediated through specialized programs that handle such conversations.

Therefore,

Different software should be available that is suitable to the style of conversation. Do not force a slow-paced letter writer to suffer from auto-completion or spelling correction while he or she is typing: Doing so distracts the writer from thinking about the message. Provide fast-paced messagers facilities to communicate quickly with as few impediments as possible; in fact, try to provide shortcuts to try to assist the writer in getting the message across rapidly. Also be aware of the intrusiveness of the message delivery system, so that people awaiting a quick turnaround are sought by the system while those waiting for a slowly written letter are interrupted when they might be working or resting.

* * *

Physical mail requires the recipient to visit a mailbox or post office. Physical mail is inherently a slow-paced medium, and so the slow pace is part of the experience of the exchange. When an email program detects or is informed that the user is writing a deliberate letter, it may make sense for the email program to suspend intrusive attention-grabbing behavior, while in the opposite case where a quick exchange is underway, notifications and the like would be welcome and should be enabled, either automatically or easily.

Edit

Last edited December 26, 2002
Return to WelcomeVisitors

Slow Letter

Slow Letter --- *RichardGabriel Nov 2001*

. . . you have determined that a textual electronic communication system will be used for slow, deliberate communication rather than rapid-fire interactions . . .

- * *

The rhythm on communication will be slow, the writer will want tranquility while writing, interruptions may not be welcome. When writing a deliberate letter, there will be many false starts, but the letter will be thoroughly reviewed before being sent; it may take days to write the letter, and part of it may change or be moved about.

The letter to be sent is not urgent though it may be eagerly anticipated. The letter may be in response to a long one received earlier.

Therefore,

Create a writing context in which automatic spell and grammar checking is turned off or can be invoked only; cause notifications of communications to be silent, quiet, or visually subtle. Enable the writer to view easily the letter to which he or she is responding, and create a way for the partially written letter to be saved for further revision and writing.

- * *

There is no need to send the letter quickly, so standard email is adequate. Spell and grammar checking should be available, but activated only on command. The visual context of the writing area---including toolbars and menus---should be tranquill, unobtrusive.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Quick Chat

Quick Chat --- *RichardGabriel Nov 2001*

. . . you have determined that a textual electronic communication system will be used for rapid-fire interactions . . .

- * *

Interaction is quick and perhaps immediate. Presence is required. Abbreviations are appropriate and appreciated on both sides of the communications. Interruptions for break-in communications is welcomed because it might be the completion of a thought, or a new topic.

It is appropriate to provide as much assistance as possible because the manner of transmitting the information or even the words themselves are not important. A message might consist of acronyms and emoticons only, or it might be a quickly dashed off note. Not much thought will go into the note, and speed of composition and rapid response are paramount.

Therefore,

Create a system which aggressively assists composition. Make spell and grammar checking immediate and intrusive. Provide easy access to a dictionary of acronyms and emoticons, and even create a system that encourages them. Make any notifications immediate and impossible to miss.

- * *

Sometimes the best mode for such communications is a chat, where there is not much time, space, or tolerance for long messages, and long pauses are taken as lack of interest or a physical interruption. Make the toolbars and menus easy to understand, gaudy even, if that makes a difference in how quickly it can be interrogated.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Ease Of Shifting Modality

Ease of Shifting Modality, e.g. phone

no versions written yet

Note: An example is when you are emailing and decide to go chat, then you would want to capture the chat in the thread of the email.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Quality Of Conversations

Quality Of Conversations

no versions written yet

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

When Our Machines Act For Us

When Our Machines Act for Us --- *RonGoldman 4 Jan 2002*

. . . in the normal course of affairs we do not want our machines trying to interact with ourselves or other people---PeopleTalkToPeopleMachinesTalkToMachines---but sometimes we need them to act in our stead. . .

* * *

There are many demands on our time and we can not always do everything that we need to in a timely fashion.

Sometimes we are physically away and if someone tries to reach us electronically we would like our machines to record their message and notify them that we have Gone Fishin' (AwayMessages).

It is easy to forget to pay a bill on time. Having our computer automatically pay our bills before they are due means one less problem we need to be concerned with.

Calendaring software now exists that lets people view their colleagues' current appointments and schedule new meetings. Some let others add to your calendar directly, while others require you to approve any additions.

On a more mundane level any task we routinely do with our programs is a candidate for automation. For example filing incoming mail into the appropriate folder is a task that modern email clients can do for us---sometimes well, sometimes with difficulty. When we are editing text it is often useful to define a "macro" that the computer can repeat as needed.

Any time that we can describe an activity such that our machines can handle it for us frees up precious time that we can devote to activities we really care about.

Therefore,

Identify places where our machines can act for us and make it easy for a person to define when and how they should act.

* * *

Whenever we delegate part of our lives to our machines it is very important that they do not act outside of the bounds we set. We must remain Safe (SafeToUse). They must respect our concerns about our personal privacy---PrivacyGradient.

For our machines to act for us requires that they have GoodIntegrationWithOtherTools.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Work Styles

Workstyles---how a person organizes their day---shifting tasks---mobility

no versions written yet

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Conveniences For Fast Work

Conveniences for Fast Work

no versions written yet

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Mail Merge

Mail Merge

no versions written yet

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Universal Access

Universal Access

no versions written yet

Note: Be able to get email wherever we are

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Information Display And Management

Information display and management

no versions written yet

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Grouping Or Display

Grouping/Display

no versions written yet

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Unify Email And News

Unify Email and News

no versions written yet

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Morning Paper

(Morning Paper (was Digest) --- *RonGoldman 12 Jan 2002*

. . . the RhythmOfConversation varies considerably for GroupDiscussions, some people will prefer to respond immediately to new messages while others will favor dedicating a chunk of time to the discussion.

* * *

Some people work better when they can devote a continuous period of time to a task; they prefer to deal with the task in one big chunk rather than lots of little ones.

An example is how people read the newspaper: Most people prefer to sit down with their morning paper and read it through in one sitting. Likewise for those who read the evening paper after returning home from work.

So with email, many people would do not want to receive dozens of messages throughout the day. Instead they would prefer to get a single digest of messages each day for each mailing list they subscribe to. Then rather than being continuously interrupted, they can focus on all of the messages on the mailing list at one time. Moreover if the digest is sent late at night, it will not interrupt most people; instead it will be sitting in their in basket when they start their day---just like a morning paper.

Therefore,

Make it possible for people to receive all the messages sent to a mailing list during the day in a single digest message. Let the user reply to an individual message in the digest, so as to respect the threads of individual conversations (ConversationsThreading).

* * *

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Mail Folders

Folders

no versions written yet

note: can file a message in multiple folders --- *unlike real world where an object can only be in one place at a time...*

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Machine Issues

Machine issues

no versions written yet

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Levels Of Scale

Here are four different levels that we feel the pattern language needs to address:

1. Human-human communication - what and why do we communicate
2. Human work habits - how does that communication fit into our lives
3. Human-machine interaction - how do we achieve this communication using computers
4. Machine-level concerns - design issues for the software architecture

Traditionally, software patterns have focused only on the lowest level, if not one level lower with how to implement the design. Even at our lowest level, we do not expect that this pattern language will address issues of email protocols (POP3, MIME, IRC), performance, or other implementation concerns. Once software developers know *what* to build, they can worry about the implementation issues of *how* to build it. The reason most software is so awful to use is because the design did not adequately address *what* to build.

Edit

Last edited December 17, 2002
Return to WelcomeVisitors

Dramatic Effect

Dramatic Effect or Deep Emotion --- *RichardGabriel 28 Nov 2001*

. . . you have determined that a textual electronic communication system will be used for more leisurely communications. . .

* * *

There is a noticeable difference between a quickly written email note and one written slowly and deliberately. Good readers can feel the thought, consideration, and emotional content of a letter written under the watchful eyes of an attentive mind.

Email has an air of triviality to it---you can easily discard it or scan it as if it were a checklist. Rarely is email kept and cherished and gone back to over and over. Email is temporary, it is expeditious. It is tough to express certain kinds and degrees of emotions in one. There are few opportunities for adding the tone and intimacy of voice that even a telephone can provide, nor are there good ways to hand-write a closing remark.

Nevertheless, a well-designed textual communication system can provide excellent facilities for composing a slowly written letter, perhaps just those parts that are informational or not as deeply moving.

Therefore,

Create a system which both suggests a handwritten letter and provides facilities to beautifully compose and print the letter, which will be sent through the post.

* * *

When a person walks out to the mailbox and finds a hand-addressed letter containing a letter from a loved on, the effect can be a memory that lasts for years. The letter can be put away, perhaps pressed in a cherished book. The letter can be held and gone back to year after year in a comfortable chair. A textual communication system that insists on only electronic transmission is missing an opportunity to remind letter writers that there are many choice to have a dramatic effect or convey a deep emotion.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Stylized Letter

Stylized Letter --- *JoshuaKerievsky 30 Nov 2001*

... you are adding stylization features to an email client that will allow users to write emails using various fonts and images.

* * *

Plain text is great for quick emails, but isn't always the best way to express more subtle ideas. Authors customarily resort to using italics, bold or different fonts to convey meaning or simply make a message easier to read. However, since not every email client can make sense of the stylization encodings, some emails show up with spurious computer codes that make the letter harder to read than plain text.

Because many email clients don't support stylized messages, many authors hesitate to use stylization for fear that their message will show up in someone's inbox with odd code in it. This means that stylization features in an email client are suspect---you don't know when you can use them safely because your email client doesn't know the capabilities of the email clients you're sending messages to.

Therefore,

Give every email client a default, but customizable, mapping between stylized text and it's plain text equivalent. When a stylized email arrives, the email client will display the letter in the richest way possible, based on its capabilities.

* * *

To facilitate the creation of stylized letters, outfit each email client with stylization features whether or not their email client supports different fonts or images. Provide a default editor, but allow user's to use their own, most comfortable editor EditingEase.

Include stylization mapping suggestions with sent emails using AnnotatedMessages.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Joshua Kerievsky

I'm a programmer. I've been programming professionally since 1988. Since 1995, I've been running Industrial Logic, a company that specializes in a pattern language called Extreme Programming. I live in Berkeley, California, about 2 blocks away from where Christopher Alexander lives. I remain deeply interested in Alexander's work and its influence on software development.

Edit

Last edited December 20, 2002
Return to WelcomeVisitors

Join Us

At OOPSLA 2000, in a workshop on Patterns & Pattern Languages, several of us (?MikeBeedle, JoshuaKerievsky, RichardGabriel, RonGoldman & others) discussed what a software-related Pattern Language that focused on human needs and values might look like. We wanted something more than a collection of the low-level, how-to patterns like those typically produced in the software patterns world. We wanted an actual pattern language that would guide a person in deciding what to build, rather than how to build it. Something that would lead to designs infused with the Quality Without a Name (QWAN). Something that would help to create software systems that would enrich our lives and help us to be more human. We thought that developing a pattern language for designing email clients might allow us to make progress towards this goal.

To develop this pattern language will take a lot of work. We encourage people to join us for discussions and to help write up possible patterns and sequences. We think it might be useful to try and use various forms of email and electronic discussion as we work on the pattern language.

We have set up a mailing list on eGroups where messages and working documents will be archived. You can subscribe by going to http://www.egroups.com/group/apl4tec. We have also created this Wiki Web and welcome your comments below. -- RonGoldman

Edit

Last edited December 17, 2002
Return to WelcomeVisitors

Next Steps

I'd like to propose that the authors get together in person for a day of writing, re-writing, merging. I think we make our best progress in person. I am available for the next 2 weeks. Since my wife is very pregnant (baby release date = Jan 7, 2003), I'd prefer meeting in Berkeley. --JoshuaKerievsky

Edit

Last edited December 20, 2002
Return to WelcomeVisitors

Exemplary Patterns

Here's a list of a few of our best efforts so far.

Please add/delete/comment on which ones people should look at first.

- AwayMessages
- SharingFeelings or Being Clear or Communicating Subtlety
- OnePersonManyMachines
- InterruptionGradient

(Actually these are just the ones that all three of us have written versions of. I'll leave it to others to pick their favorites.)

Edit

Last edited December 18, 2002
Return to WelcomeVisitors

Sequences Of Patterns

Sequences:

A pattern language consists of more than just a collection of connected patterns. It also includes *sequences* that step one through the design process, pattern by pattern, from the largest design questions down to the smallest. Reading through a sequence should create a mental image of the final design. Each sequence is a story, in our case, a story of how a software application would enable a person to communicate appropriately in a particular setting. So one sequence might result in a design that fosters a sense of family or community, while another might focus on more impersonally providing answers to customer questions.

Here's an example of a possible mini-sequence (with pattern names in quotes):

> A major need of most people is IntimateCommunication with those they love and care about. Much of the time though they are engaged in "Solo Work" and any communication is a "Secondary Task". Providing a Remote Presence that enables them to know when family members are online, and that allows them to exchange "Low-overhead Messages", will reinforce the family bond. "Peripheral Awareness" makes the communication more natural. Being able to signal a "Closed Door" respects the needs of the workplace. "Automatic Connection" makes it a regular part of the computing environment.

Such a sequence might be used to generate something similar to AOL's Instant Messaging system.

Note: We need to update this sequence to use our current patterns---it was originally written before them. Or we can add the patterns it names to the pattern language.

Edit

Last edited January 7, 2003
Return to WelcomeVisitors

Ron Goldman

mailto:rgoldman@cs.stanford.edu

I'm a researcher working at Sun Microsystems on alternative software development methodologies and new software architectures, with a strong interest in user-centered design and information visualization. I'm currently finishing up a book on how companies can participate in open source software development. Prior to Sun, I developed a program to generate and manipulate visual representations of complex data for use by social scientists as part of a collaboration between NYNEX Science & Technology and the Institute for Research on Learning. I have a continuing interest in the design of programming languages and have developed various programming environments (IDEs). I have a Ph.D. in computer science from Stanford University where I was a member of the robotics group.

Edit

Last edited December 17, 2002
Return to WelcomeVisitors

Helpful System

Helpful --- RonGoldman *24 Jan 2002*

. . . when people are trying to communicate with each other they don't want to appear stupid.

* * *

There's an old saying: To err is human but it takes a computer to really screw up!

It is very easy to make mistakes, but with a computer we can all to easily embarrass ourselves before a wide audience. How many times have you seen a message sent to a mailing list that was clearly intended to be a private message to one individual? Whoops, someone did a "reply to all" that included the full mailing list. Or someone sends out a message that is clearly not appropriate for the mailing list it is sent to? Or someone is angered by a message and sends off a quick retort that they later might regret?

The Eudora email client has several ways that it tries to be helpful including highlighting misspelled words automatically, warning if a message has no subject specified, or flagging messages that use aggressive, demeaning or rude language in any email you are writing with one to three chili peppers. Both the spell checking and the flame alerting are done by discreetly annotating the message---the user is not interrupted and can choose to ignore the warning.

Therefore,

Anticipate ways that people will make mistakes and write computer programs so that they either correct or inform us in an appropriate way. Our computers should be helpful, and watch out for us. But remember that the person is ultimately in charge.

* * *

In our correspondence with others, a GentleReminder can help us avoid appearing rude by not responding.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Timely Response

Timely Response --- *RichardGabriel 2 Dec 2001*

. . . You want to communicate with someone far away but that person is hard to get a hold of . . .

*   *   *

We have friends, family, and colleagues who are hard to get a hold of. You send a letter and it's not answered; you phone and they never answer or return your call.

Some people have an aversion to phones, particularly when making a call into an unknown situation. Letters are slow to begin with, and because they are used for a slower rhythm of communication (RhythmOfConversation and SlowLetter).

But sometimes you need to get a hold of someone, and you'll do whatever it takes to get a response.

Therefore,

Send an textual electronic communication. For some reason---perhaps because it is a relatively new medium---people respond to email more rapidly than to other forms of communication. Transmission of such messages are nearly instantaneous, so there is a degree of immediacy that is interpreted sometimes as urgency, and this can instigate a quick response. Email, for example, tends to age and go stale in the inbox in a way that letters don't. Because people spend a lot of time with email, there is a potential reminder every time they go back to their email for whatever reason, while phone messages are apparent only when people are interacting with the phone-and then it is usually just a flashing light.

*   *   *

It is also cheaper, so it has a lot of advantages. For sending messages quickly see LowOverheadMessages.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Cheaper To Use

Cheaper --- *RichardGabriel 1 Dec 2001*

. . . You want to communicate with someone far away but have no money for it or don't like to spend it. . .

* * *

We all have friends who live far away. The American culture from the beginning of our country has been based on a continued and continuously renewed push from East to West, which has spread families from coast to coast. We yearn to go to school somewhere else, and we find our loves and livelihoods there and move in. So we want to communicate.

We don't like bills or paying by the minute to do something that's reflective. Why pay to sit and think between words or phrases? Or we don't like to pay, period.

Therefore,

Use some form of textual electronic communication. It's a local call, which seems to cost nothing because it's part of a monthly fee you pay whether you use it or not, or it's a connection to the Internet that is paid for by the month. It's cheap and you can spend time thinking.

* * *

It's also faster (TimelyResponse). And people tend to pay attention to email when they won't to most else.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Mail Filtering

Mail Filtering

no versions written yet

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Annotated Messages

Annotated Messages

notes

* * *

Problem: People want to be able to make notes or edit messages that they receive.

Therefore,

Do not treat received messages as static and immutable. Provide ways that people can annotate them with notes and comments. Allow the messages to be edited.

* * *

Edit

Last edited December 24, 2002
Return to WelcomeVisitors

Small Work Groups Come And Go

Small Work Groups Come and Go --- RichardGabriel *30 Nov 2001*

. . . you are designing a textual communication client and are working on how to address people . . .

* * *

The people we work with changes over time both as our interests coincide and drift apart and as projects start, mature, and end. Who these people are and when such lists come and go are not part of a foreign infrastructure but a part of our private lives. We need to easily communicate with our colleagues.

A distribution or mailing list is nothing more than an alias representing a set of individual addresses, but such things happen to be difficult to set up, perhaps because someone had decided long ago that clients do one set of things and servers do another. The same is true of buddy lists, weblogs, and archives.

Or perhaps someone has decided that only a higher authority should be allowed to enable aggregations to work together. Or perhaps existing tools are too primitive to enable people to set up easily new mailing lists, buddy lists, and weblogs.

When a group disbands, the archives might need to be shared or disposed of, and sending to demised mailing lists needs to be handle with grace and care. When people come and go from mailing lists---at their own pace and timing---no one really knows about it, nor does the software act properly and without dumb failures. For example, vacation messages go to mailing lists.

People often stick with one form of conversation because they don't know another is available. For example, it is not easy to switch to a chat from an email discussion, and a person might not even know that all or most of the interested parties are online and could engage in more direct communication.

Mailing lists sometimes grow too large and need to fragment based on local interests, but tools are unable to help with this.

Therefore,

Make it easy for a regular user to setup and disband a small work group "mailing list" or virtual working space. Offer a small collection of tools that allow the group to interact in different ways. Partition the functionality so that different bits of it can be combined in appropriate ways.

Make it so that mailing list discussions can morph into chat rooms or instant messages, that archiving across discussion formats is continuous, that people can easily update their own best addresses without intervention by strangers, that distributions lists can be easily combined or split up, that temporary additions and removals can be done easily, and that control of who can talk to whom is handed back to the people who want to talk and not system administrators.

* * *

Archives, annotations, and flagging need to be considered and blended with the client. The entire state of the client, address book, and mailing list state of a client must be easily maintained and transparent. Typing needs to be minimized in such maintenance.

Small Work Groups Come and Go --- RonGoldman *9 Feb 2002*

. . . the nature of GroupDiscussions is that they start whenever several people have something to discuss and that they eventually end.

* * *

As people we belong to many different groups. As our interests and projects change, so do the groups we care about, at work, at play and at home. Members of each group need to converse with each other and so need ways to get together either in person or through some technological means.

In the physical world a group discussion can occur just by several people getting together. Sometimes this requires advance planning such as choosing a time that everyone is available or reserving a room to meet in. For conference calls each participant must agree to a time and then call a prearranged number. Many group discussions occur spontaneously---a bull session in a hallway---or as a side-effect of other activities---sharing a lunch table in the cafeteria.

In the electronic world group discussion can be more problematic. All interaction is mediated by some technology that must first be mastered. The initial hurdle is to setup an electronic meeting space for the new group. This requires having access to tools not to be found on most personal computers that can support creating group mailing lists or chat sessions.

For an online chat session the group needs to locate some type of chat server that they can use to create a possibly private chat session, such as a chat room at some web site, an Internet Relay Chat (IRC) channel, an Instant Message service, or whatever.

For a mailing list a computer that allows the creation of new mailing lists must be available. Then everyone interested needs to add themselves to the list, or maybe the mailing list creator can sign them up.

For a newsgroup there is a somewhat lengthy process required to get approval to create an official new newsgroup, though "alt" groups can be set up at any time. Once created there is no guarantee if and when various news servers will carry the new group (especially one in the "alt" universe). Creating a new newsgroup may be too much overhead for a group with a limited life span. It may make more sense for short-lived discussions to use an existing newsgroup.

In addition to a channel for conversation, groups will want to know who is participating in the discussion and may want to prevent strangers from joining in. Moderated mailing lists and private chats can be used to preserve privacy. Most chat systems have commands to display current participants.

It is also very useful to have some way to preserve discussion for people to review or search later.

Some groups continue for long periods of time and may better be considered to be communities. Other groups disappear after some time and there should be a way to let people know that the group is no more.

Therefore,

Design electronic communications tools to support the entire lifecycle of group discussions. Make it easy for people to start a new discussion, be it via chat, email, newsgroups or the web. Plan for people to join and leave the discussion after it starts. Design a way for the group to gracefully end the discussion.

* * *

An Archive (KeepArchives) can provide a history of the discussion. People often are engaged in multiple conversations so Threading (ConversationsThreading) can help keep the different discussions distinct. It should be EasyToUnsubscribe.

Edit

Last edited December 24, 2002
Return to WelcomeVisitors