Cross-channel Mobile Social Software: an empirical study

Clint Heyer[1,3]**, Margot Brereton**[2,3]**, Stephen Viller**[1,3]

[1] School of ITEE,
University of Queensland
St Lucia
Brisbane, Qld 4072, Australia

[2] Queensland University of
Technology
PO Box 2434, Brisbane
Qld 4001, Australia

[3] Australasian CRC for
Interaction Design (ACID)
PO Box 2008, Kelvin Grove
Brisbane, Qld 4059, Australia

clint@itee.uq.edu.au, m.brereton@qut.edu.au, viller@acm.org

ABSTRACT

In this paper, we introduce a prototype system designed to support mobile group socializing that has been appropriated for everyday use by 150 users over 18 months. The system supports cross-channel communication, allowing users to participate in group conversations using text messaging, instant messaging, email and the web. It does this with the "console," a uniform text-based syntax that enables the prototype to be used over a variety of mediums.

We found that participants used the system mostly for ad-hoc coordination rather than chat, with pervasive, cross-channel group communication supporting an informal "half-invite" style of invitation. We examine why coordination dominates over chat, suggesting that cross-channel mobile group messaging serves a distinct role, different to traditional text messaging, instant messaging and email. Furthermore, we found differences in the content and usage habits across channels, for example messages sent from a computer were more likely to refer to time and location than those sent using a phone. We also discuss the usage of the prototype and compare it to other work.

Author Keywords

Text messaging, multi-modal, multi-channel, group messaging, field trial

ACM Classification Keywords

H.4.3 Communications Applications, H.5.2 User Interfaces, H.5.3 Group and Organization Interfaces.

INTRODUCTION

Mobile Social Software (MoSoSo) identifies systems that can be used whilst mobile and aim to either directly support socializing or indirectly take advantage of social information or social networks.

The challenges of deploying a mobile social system are threefold:

1. **Utility:** the system must facilitate socialization in a way that is more valuable than existing systems, so that it is worth using.

2. **Accessibility**: the system must be accessible using a wide range of devices and media in order to enable as many people as possible to participate, thereby further increasing its utility.

3. **Usability**: the system interface must be simple to use so that people can easily integrate it for everyday socialization.

Multi-channel systems that work seamlessly between users across web, email, chat and phone messaging have the potential to provide better support for social networks. However, it is challenging to design a simple interface that is easy to use that works across different media and devices (in order to maximize accessibility and utility).

This paper describes "Rhub", a multi-channel communication system that was developed and deployed iteratively, with use and user feedback guiding design decisions and iterations. Our main goal in developing Rhub was to explore how small social groups communicate, coordinate, and share, and how a prototype system offering basic cross-channel communication and group maintenance support might be used to support these activities. Rhub was appropriated for use by 150 users over 18 months. The paper focuses on three primary features of Rhub that are key to its multi-channel capability, which also relate to its utility, accessibility and usability: group messaging and its use; use of presence awareness features such as locations; and design and usability of a text command syntax for group messaging across multiple media and devices. Trade-offs made in order to maximize the triad of utility, accessibility and usability are discussed. Methods used to gather and evaluate data about use are also discussed.

While commercial mobile social systems measure their users by the thousands or millions, they are relatively closed to researchers. Research systems, on the other hand, are open yet typically run for a short period of time or with a small number of participants. As such, it is difficult to determine how the technology "settles" or is appropriated within the given social context or the nature of usage when novelty effects diminish. The primary contribution of this paper is discussion of the deployment and actual, long-term

usage of a mobile social research system with a large number of participants.

RELATED WORK

Early MoSoSo work utilized wearable computers as they offered higher levels of sophistication than the days' mobile phones [10, 14], however most recent work uses client software installed on a mobile or that can be used via text messaging. There are difficulties with both approaches: client software requires maintenance, and there are a multitude of platforms to support, while text message-based interfaces can be difficult to use because of the lack of transparency and visual cues [17]. Whilst MoSoSo is a broad field, this discussion focuses primarily on systems that support social communication.

Context-oriented systems

Some systems use context-derived data in order to determine when a message should appear, for example showing a message when someone arrives home [8], or sending a message to everyone at particular location, in effect determining a dynamic messaging group, as in Dodgeball (http://dodgeball.com) and Playtxt (http://playtxt.net). While the aforementioned services use predefined named locations, Loopt (http://loopt.com) references location using GPS-supplied geographic coordinates. Coordinates are accurate; however they lack semantic value for a person unless they are displayed on a map. Reno [19] attempts to link meaningful, user-defined location names automatically to current location using cellular tower-derived position information. It was observed that Reno users mostly relied on pre-populated locations set by the researchers rather than creating their own. Misunderstanding can result from location-based systems as they only capture a narrow aspect of context and automatic disclosure of location can be both beneficial or detrimental, depending on the broader context [19].

Presence and status

Presence sharing is a common use of text messaging [12], and there are services which attempt to make this type of context information easily disseminated, such as Twitter (http://twitter.com) and Jaiku (http://jaiku.com). Presence notifications are textual and free form, for example using the message *"I'm at working having a dull day"* rather than setting a location to a specific, pre-determined place or coordinate. Notifications can be posted to and from a variety of technologies such as the web, SMS and instant messaging (IM). Free-form presence messages allow the sender to imbue messages with their own sense of meaning at the expense of opaqueness from the systems' point of view – it is not straightforward to place someone at a geographical coordinate from arbitrary text.

Text messaging

Text messaging, or Short Messaging Service (SMS), is a popular mobile communication system used mostly for person-to-person messaging. Popular with teenagers [7, 15], text messaging is seen as a low-priority channel of communication because of its cheapness and asynchronous, background nature [9, 11, 16]. Regarding the kind of tasks that SMS is used for, Ling's study of 882 messages sent by Norwegian teenagers [12] reports that only 33% of messages are about coordination, but when related categories are included ("commands or requests", "questions", "answers", "information") the total is 64%. On the other hand, Grinter and Eldridge's [5] study of 185 messages from British teenagers show that 55% of messages were coordination-related.

Group messaging

Text messaging was not designed for group usage. If a single text message is sent to multiple recipients, each is unaware of who else it was sent to (or indeed that it is a group message), making group replies difficult. There have been various studies which explore the possibilities and usage of more sophisticated group text messaging systems [2, 3, 6, 18]. Rhub is similar to "Swarm" [3] in this respect, however Swarm is only used for group messaging, and only supports usage via SMS. "Slam" [2] can additionally be used via client software installed on a Smartphone, and also supports sharing of photos. Swarm was deployed for a long period with a small number of users whilst Slam was deployed for a short period with a small number of users. Reported usage of group messaging systems varies. With Slam, 48% of group communication was chat, 18% coordination, quite different from the results of Swarm, which was mostly used by one large social group (26 members) in which 68% of messages were coordination, 28% chat. With reported figures ranging from 18% to 68% of messages being for coordination, it is worthwhile to investigate the area further.

PROTOTYPE: RHUB

Our original goal was to explore how small social groups communicate, coordinate and share. We started with a basic system that supported messaging, discussions, user profiles and group management. Once the system started being used, observations and feedback from participants were reflected upon and used to inform further development or redesign in an ongoing, rapid, iterative process. Rather than only refining what was already present in the design, we also experimented and provoked response by adding new features to see how they would be used. In the first three months of deployment, micro-iterations happened frequently during the day, with this process slowing as the design matured.

To maximize the pool of potential participants, we built our prototype to use existing, mundane technologies such as the web, IM, SMS and email. These technologies are already appropriated into people's everyday lives and are readily accessible for use. Leveraging existing technologies also allowed us to run the study for a relatively long period as participants used their own software and devices. Importantly, this also meant that the study could be run "in the wild" outside of the confines of a laboratory study with people using it for everyday purposes as need dictated.

To ground the later results and discussion, we will introduce the three pertinent features of Rhub: console, messaging and locations; however it also supports other common social software features such as sharing of photos, tagging, and aggregation of RSS-based feeds. While Rhub is not overly technologically complex, it is different from existing academic work primarily in that it was designed for cross-media communication and has a flexible location and presence system. It also makes use of many common social software idioms and features that allow participants to have a richer experience, and thus encourage further participation in the study.

Console

To allow Rhub to be used across a variety of mediums, or channels, we developed a command syntax, similar to a command prompt, which can be used in conjunction with the primary rich web interface. We call this syntax the "console", and it is used by sending text in the appropriate form using a supported medium (Table 1). The same commands can be sent using the web, SMS, IM, or email although system responses may vary depending on inherent limitations of each medium. The advantages of this approach are 1) the system is completely server-side: we do not need to maintain software for different mobile and desktop platforms, and users do not have to install anything, 2) users with entry-level mobile handsets can participate, and 3) users' existing norms and knowledge can be leveraged. For example, rather than requiring our participants to carry around a new device, we leverage the fact that they usually already carry around a mobile phone.

There are also many difficulties with using such an interface, primarily that there are no visible cues for the interface for the user to determine the state of the system, its available commands or options. Although this form of interaction may seem cryptic, once mastered, the user can participate fully in messaging across media. This approach emphasizes robustness of basic function over initial ease of use and might seem to be antithetical to a user-centered approach. This is a valid criticism, however we feel this approach is worthy of exploration, in particular, 1) because text messaging is not itself very user friendly by a number of measures, and yet because of the utility it offers, its uptake has been enormous and 2) in our experience of uptake of systems, as long as they offer value and are robust (a primary user need), users seem willing to learn them and to develop and support their use through their own methods and workarounds such as post it notes or saved messages to remind them of key syntax and finally, 3) there are gains for the user in often unacknowledged areas such as the ability to use the system on any phone and the lack of need to configure software. That said, we aspired to develop the simplest text based system possible, through iterative user centered design, in order to support intuitive use and to minimize the need for user workarounds.

We did not anticipate all of Rhub's functionality to be used via the console, only functions most likely to be used when

Prefix	Metaphor	Command	Example
>	Send to	Messaging	*>Johan: Hello*
			Sends 'Hello' to Johan
@	At	Presence and locations	*@cafe?*
			Finds people at the café or nearby
!	Command	Manipulating settings, or other tasks	*!info Carsten*
			Returns Carsten's contact information
&	And	Group related	*&tennis add Lisa*
			Adds Lisa to the 'tennis' group

Table 1. Console syntax overview. The syntax allows Rhub to be used from a variety of channels such as SMS, IM and email.

the main web interface was unavailable, such as when mobile. After identifying the main functionality to expose, we attempted to group the features and assign commands in a way that was discoverable, memorable and usable.

Messaging

There are three main forms of messaging within Rhub, person-to-person (direct) instant messaging, group instant messaging and threaded discussions. Messages are cross-media in that they can be sent and received using the web, email, SMS, or IM (MSN or Jabber) seamlessly. Messages are also persistent, in that they are available on the web for later perusal, regardless of the originating source.

For example, in the one group conversation, Thom might use IM because he is in front of his computer, Phil might participate via SMS because he is mobile and Jonny might correspond via the web. Rhub ensures that every message sent to an entity, regardless of source, is distributed to all concerned parties, regardless of destination. Thus, the conversation spans technologies as each person sends messages using the medium that best suits them, and receives messages using an appropriate medium. When sending a message using Rhub's website users can choose which channels to use for sending a message. By default, sending a message using the console will attempt all mediums. To decide which medium to use when contacting someone, Rhub uses a series of heuristics based on which mediums are available for a person, which mediums the message has been permitted to use, and which mediums the user has allowed for a particular message type. For example, if you have permitted Rhub to send you group messages via IM, SMS and email, and your IM status is "online", you will get a group message that way, but if you walked away from your computer and your IM status changes to "away", you will receive it via SMS. If the sender thought the message was not very important, they might choose to have the message delivered via online means only, and in the previous example, the message might be sent via email instead of SMS.

To illustrate further, Lisa might send a message to the group "tennis" with this IM, using the console syntax:

>&tennis: Anyone want to play tennis this afternoon?

Members of the tennis group might receive the following message via SMS (messages for other mediums are formatted and worded slightly differently):

Lisa: Anyone want to play tennis this afternoon?
To: &tennis (to reply, txt '>&tennis [msg]')

Carsten, a member of the tennis group, might then reply using the full syntax or simply a plain text message:

yes, I'd be interested after 1pm

Rhub automatically routes it to 'tennis', because that was the last entity he got a message from.

Groups

Groups within Rhub are designed to be lightweight: quick to create and easy to manage. Groups have a variety of settings to set the visibility and openness of the group as well as set who can send messages to the group and how. In addition to instant messages and discussions, groups can have associated locations and sets of things (such as photos, bookmarks and blog entries). By default, a group is publicly viewable and joinable, and all group members can send messages to it using any available medium.

Locations

Locations in Rhub are mostly used for presence features, such as notifying friends when they are close to each other, or displaying users on a map. They can also be used as context for discussions, for example talking about upcoming events at a music venue, and instant messaging, for example sending a message to everyone who has set their location to a particular place.

Rhub uses flexible referencing of locations that allows users to refer to precise "well-known" locations by name, which have additional encoded data such as geographical coordinates, or set arbitrary free-form locations, which may mean something to other users, but not to Rhub itself. Unlike other work, Rhub also allows users to add and edit existing locations (which can then be used by others) and set personal aliases. Thus, instead of having to SMS *"@the royal exchange hotel"* to set location, it is possible to send *"@pub"*. Rhub resolves locations by looking for an exact matching title and if that fails, it looks for an alias the user has defined, or the user's friends, or finally, any matching alias. Aliases allow users to refer to locations in a natural way, and since they can be inherited from others, they can be used without a "setup cost" such as in Reno [19].

METHOD

We used a multi-method approach to understand system usage and explore participants' perceptions of the system, which is outlined in this section. Quantitative data was collected in the form of extensive automatic logging of usage; qualitative data was gathered with semi-structured contextual interviews, Rhub-delivered quizzes, informal conversation with participants and a workshop.

Prototype

We initially invited 19 friends and colleagues when launching the system in February 2006. In 12 months, the number of participants grew to 108 as the authors and other participants invited further users. Users are asked to sign an informed consent form the first time they log in to the website; messages from users who did not give consent are excluded from this analysis. While we did seed the network with our own friends, because of friends inviting their friends and so on, the system had many users not known to us personally.

The busiest Rhub group, in terms of messaging and members who actively used the system was "Alpha" (names changed for anonymity), created for a university sporting club by one of the authors, who was also an active member of club. Rhub was also introduced to other groups; however, this large, dynamic, highly social group was the most prolific. Membership size of Alpha varied during the course of the study, the majority of the club's members are foreign exchange students and typically return home at the end of each semester.

Content of 500 randomly selected messages from Alpha were manually analyzed by one author who was a member of Alpha, using Farnham and Keyani's ontology [3] as the basis of our analysis. Messages were labeled with one or more categories, for example, a single message proposing a dinner plan might be labeled with coordination-invitation, reference-location, reference-time. Although analysis would benefit from being double-blind checked by another researcher, we had to abide by the privacy agreement that only the first author would analyze messages. It was decided to cross-check data by triangulating with other methods: interviews, quizzes and workshop.

Interviews

In-depth interviews were conducted with fifteen participants, selected based on availability, all of whom were known to the researchers. We conducted the interviews at a place of convenience for the participants (typically their home), and used printed transcripts of Rhub activity and the user's own phones and devices as scaffolding for the discussion. Interviews took approximately an hour with audio of the interviews and notes recorded on a tablet computer and later transcribed.

Quiz

The quiz we ran using Rhub has two purposes. One, to gain an understanding of our users' contexts and technology usage and two, to explore the utility and effects of using a system like Rhub for research. We use the term "quiz" rather than "questionnaire" to convey its informal, lightweight nature. Questions were delivered using Rhub to members of two groups (Alpha, and 'Iota', a group of academic colleagues), and thus could have been received by or replied to using SMS, IM, email or the web. In the spirit of cultural probes [4], the questions were designed to help understand context and practice rather than glean

quantitative data. Questions (16 in total) were organized into six themes: location, activity, presence, technology availability, technology use and technology preference. For example, one question from the activity theme was: "*what's your position: walking, reclining, jumping, sitting etc.?*" The quiz took place over 80 days, and we received 102 responses from 590 sent questions (17% response rate). Questions were answered quickly, 37% within two minutes and 69% within 10 minutes, suggesting that answers are about the current context rather than a recollection of a past. We found this a useful method for mobile contextual inquiry and framing of quantitative results.

Workshop

We also ran a workshop, inviting members from the Alpha group, and four attended. The workshop ran for two hours, and a series of design activities were conducted, aiming to explore users' understanding and use of Rhub. We used a combination of exercises which included sketching, a variation of the video card game [1], annotation of past messages, and reflective use.

While the workshop and quiz results were helpful in supporting our understanding of Rhub's use, extended discussion of the quiz and workshop methods and results are beyond the scope of this paper. We have focused on results from interviews and quantitative data here.

RESULTS AND DISCUSSION

In this paper, we briefly outline the results of our preliminary data analysis. Generally, Rhub's group messaging feature was the most used, typically with participants organizing last minute, ad-hoc events. People also used it to share amongst the group and wider Rhub community, such as adding favorite shops using the locations feature, uploading photos of events and establishing social ties using the contacts system. By July 2007, 41 groups had been created, 137 locations defined, 891 contact relations made, 950 photos uploaded, 1,942 tags applied and 1,962 group messages sent.

Messaging

Participants used Rhub in a sustained manner over a long period, indicating usage was not due to novelty (Figure 1). Much of our discussion centers on the usage of Rhub by the largest single group, Alpha. As a result, the generalizability of our findings may be limited; however, we feel it is still a good illustration of how a group with existing social ties can use a mobile social system. Alpha, over the course of the study, averaged 18 messages per week with an average of 30 members. The mean weekly message rate for all groups was 1 (standard deviation 4.9).

We noticed that people's usage of Rhub reflected the channel they received messages on. A group ('Iota') of academic colleagues who were mostly all receiving messages via IM sent frequent short messages in quick succession while Alpha tended to have a slower pace and denser messages. Iota's high-throughput usage managed to
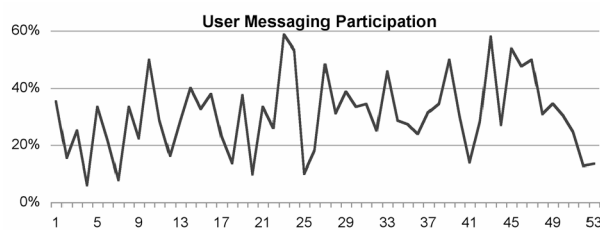

Figure 1. Percentage of active users who sent a message, per week. The spike at week 23 corresponds to the FIFA World Cup, the trough at week 25 with a system failure while the authors were away. 108 users represented by week 53.

crash our messaging system, while Alpha's low-throughput messages did not, even though Alpha sent a much larger quantity of messages over a longer period of time.

Message goals

Participants utilized messaging in several ways within Rhub. Analysis of 500 random messages sent to the group Alpha shows that participants mostly sent messages for coordination (Table 2).

Social bonding (messages that have no particular coordination or organization goal) was less prevalent (30% of sample) than coordination messages (70% of sample). Coordination messages were further broken down into subcategories. Frequently, an invitation or idea is sent: "*whos getting funky 2nite? I finish work soon!*" (28% invitations) which is usually followed by group negotiation of a time and location. During the course of this negotiation, some will indicate whether they will come (17% commitment); query aspects of the plan (18% questions); direct others on a course of action "*[…]RSVP me before Wednesday!*" (6% directives) or suggest alternative plans. While the event takes place, members might send messages to report on how it is going (12% reports), often with an aim to entice more people to come: "*Cookamungas anyone? it's got people!*". Messages might also promote or hype an upcoming event (4% buzz-building), for example this response to a wrestling themed party invite: "*Hulk Hogan is back in fashion baby.*"

We also looked at references contained in messages, identifying references to location (25% of messages), time (24%), people (13%), status (11%) and activity (23%). Half of all messages sampled contained a reference of some type, predominantly appearing in coordination messages.

Family (no. of messages)	Categories (% of messages within message family)
Social Bonding (153)	(No significant subcategories)
Coordination (351)	Invitations (28%), reports (12%), questions (18%), directives (6%), commitments (17%), buzz-building (4%)
References (249)	Location (25%), time (24%), people (13%), activity (23%), status/presence (11%)

Table 2. Content analysis for 500 random messages from group Alpha (one message may be tagged in multiple categories, so the percentages do not sum to 100%)

(235 out of 351, or 67%) rather than social bonding (12 out of 149, or 8%). Messages were tagged as a status reference if the sender included information about their current state or context, for example *"Short pub crawl? Exam tmro but i'm sooo bored"*. Status was often included as an 'excuse' when declining invitations, or to otherwise fill in context for others. Not surprisingly, activity was often referenced in coordination messages, for example suggesting a potential event: *"I'll be well keen in three hours for drinking and or football"*.

Nature of usage

Coordination over chat

We found that Rhub's messaging capability was mostly used for coordination (70%), rather than social chat (30%). This was also supported by interviews we conducted with participants, who saw Rhub as being primarily a tool for coordination, with some actually suggesting that Rhub be explicitly *not* used for chat. This result is very similar to the Swarm [3] group text messaging system (68% coordination) suggesting that the finding might be generalizable across individual social groups. The high level of coordination within the larger groups in both Rhub and Swarm contrast to the higher level of chat which took place in Slam [2], which had smaller groups. Coordination between group members happened quickly over a burst of messages, a form of micro-coordination [13]. Participants were active in shaping messaging norms using both talk and action [20]. Users who sent too many messages to a group were publicly reprimanded by others, both in person and also using Rhub itself, such as this message sent by Michael to a group that he thought Sam was sending too many messages to:

"Sam…your Rhub rights have been temporarily revoked…feel free to start again in 2 months time…"

Existing handsets are not usually designed for composing, managing or displaying group text messages, and as such usability suffers. Additionally, a phone's message arrival notification does not distinguish between group or private text messages for systems such as Rhub, which makes it difficult for users to determine the urgency for reading a message. These two design factors—coupled with the fact that group text messaging increases the quantity of received messages—results in a higher level of disturbance than direct messaging. Participants reported the extra quantity of group text messages they received was only annoying if the messages were of no value. Coordination messages were acceptable because usually the event was something of interest. Even if they were not interested, they realized that next time they might be, therefore the annoyance now is tolerable, considering potentially useful messages later. As one interviewee stated, *"that's the price you pay for group messaging"*. Chat messages on the other hand were seen as extraneous and better sent using less intrusive systems such as IM or a web-based forum.

In-between email and text

Interview participants likened Rhub to be something in-between email and text messaging. Email was seen as a cheap, reliable way of distributing a message to a group of people, but interviewees did not consider it useful for short-term organization because many of their friends did not check email regularly. Text messaging on the other hand was thought to be quick and instant, but unwieldy for group messages. For our interviewees, Rhub was seen as a more robust way of getting a message to a group, easier than text messaging and more direct than email. Participants discounted IM for group coordination because of the difficulties in synchronizing availability. Most people valued Rhub for keeping them "in the loop" of their group's social events and happenings. When they themselves had something to say or event to organize, they found Rhub highly useful for "getting the word out".

Half-invites

We observed a style of off-the-cuff, ad-hoc form of coordination that group mobile messaging supports particularly well. Unlike traditional invitations that require response, "half-invites" tell a group of people that an event is taking place and others are welcome to join, or as one interviewee stated, "we're doing this, come show up". Half-invites often invite implicitly, for example this message sent on a Sunday afternoon:

"Sunday night at the pub: because you know you shouldn't".

Within the Alpha group, we often observed small groups of two or three people deciding on an event amongst themselves—such as going to a particular pub—with one of the organizers sending a group Rhub message to entice others to attend. For these types of informal events, responses are not required, and thus invitations are easy to ignore if they are of no interest. For small groups that would not mind additional numbers, sending a mass half-invite is quick to do, and may yield some extra company.

Half-invites suggest an informal, inclusive style of events where attendance is fluid and invitees may not be well known. Rhub is particularly useful for this, as contact details do not need to be known or collated by the inviter—they can just send the invite to the group, and trust Rhub to route the message appropriately. Whilst it may seem odd that people would want to invite others they do not have contact information for, we found it was a common case with Alpha group members, as one interviewee said of the scenario: *"I might not know your number but you're invited"*.

Participants were also aware of the less personal nature of group invitations. They felt that mass invitations were less personal than individual invitations (even if they were delivered using the same technology). Rhub groups are inclusive, and it is not possible to selectively message within a group—it is either a message to the entire group or a message to each person individually. Natural social
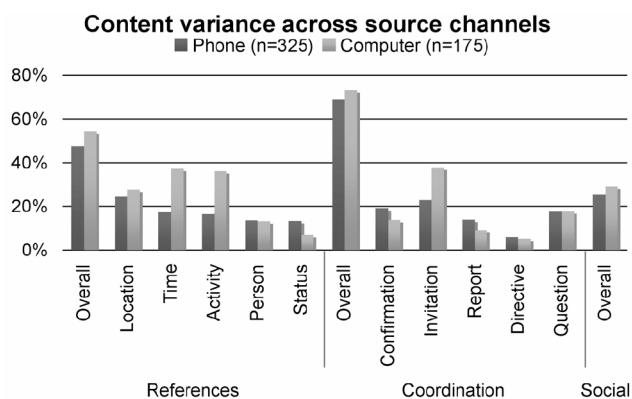
Figure 2. Message content across phone (SMS) and computer (IM, email and web) sources. Percentages show the number of messages categorized out of the total number of analyzed messages for that source, such as 24% of analyzed messages from a phone contained a reference to location.

tensions arise and there might be events that you do not want *everyone* invited to, or you may not want particular people to be privy to the group's messages. We found that messaging dynamic subsets of groups was not a common enough scenario to support within Rhub. Participants either manually messaged a set of users, or for less transient sets, established a new group. For example, five offshoot groups were established for sub-interests within the Alpha group membership. Rhub offers privacy and moderation controls, it is possible to have secret groups which are invisible to non-members, or public groups that require invitation to join, for example. While it did not happen during our study, we would expect that forcibly removing someone from a Rhub group would result in significant social ramifications. Two users requested to be removed from Rhub entirely after their relationship with their respective Rhub-using partners ceased.

Fixed versus mobile usage
To understand differences between fixed and mobile usage of group messaging, we analyzed message content with regard to source channel (Figure 2). Instant message, email and web messages were grouped as "computer" (175 total) and SMS messages were grouped as "phone" (325). We generalize that messages sent from a computer are likely from a fixed location rather than highly mobile. Overall, there is little difference in the percentage of messages that contain references or relate to coordination or social activity. Mobile-sourced messages however contain less reference to time and activity, perhaps as they are "in-
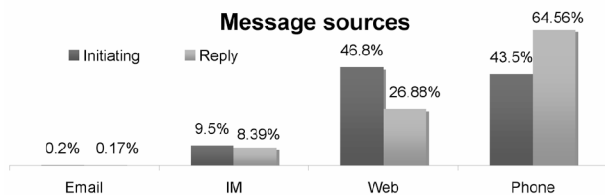

Figure 3. Initiating and reply message sources. 64% of all replies were sent using SMS while the web was used (proportionally) more for sending initial messages.

context", in the midst of activity, whilst fixed-location usage relates more to future activity where time and location must be defined. Invitations were more likely to be sent using a computer, echoing data showing initial messages were more often sent using the web (see later discussion). A slightly higher percentage of coordination confirmation and report messages were sent from a mobile.

Cross-channel communication
How effective or useful is cross-channel communication? Rhub combines channels for both controlling the system as well as for receiving messages from others. First, we examine how cross-channel system interaction was used, that is, user input to the system and secondly we discuss the implications of cross-channel communication as an output of the system.

We processed group instant messages using a basic algorithm, grouping messages into conversations, based primarily on the temporal difference between subsequent messages. Applying this algorithm on 1,977 messages (78 weeks) produced 496 conversations. We found that conversations tended to happen within a single channel (63%) or a pair of channels (29%), with only 7% taking place over three channels and no conversations taking place over all four supported channels. We believe this to be a reflection on the ubiquity of mobile phone access (most messages were sent and received using a phone). After receiving a Rhub message on their phone, users found little reason to go to a computer to compose a reply message: 64% of replies were sent using a phone, while only 27% of replies were sent using the web yet the percentage of initiating messages for both mediums is similar (Figure 3).

Console: text-based interface
In co-opting existing mediums for an alternative purpose, such as using SMS for system interaction when it is normally used for human to human conversation, we inherit usage norms from the medium as noted in [17]. Originally, the console's parser was strict: to send a message to a group using an SMS, the user had to use a precise syntax of *>&[group]: [message]*, such as *>&tennis: Game this afternoon*. This was to ensure the correctness of the parser and to avoid miss-sent messages. Even though all messages sent by Rhub include a postfix hint message *"To reply send >&tennis: [msg]"*, we found that many users tended to instinctively reply as they would to a person, without any special syntax. As a result, we successively made the parser less strict, for example dropping the requirement for the ampersand to denote groups and the colon to separate group name from message. Input that cannot be parsed is treated as a message destined for the entity the user last got a message from, within a three-hour period. Thus, after receiving a group message, the user can enter some text and reply without having to add special syntax and Rhub will route the message to the appropriate entity.

When initiating a conversation using Rhub, some interviewees reported they would turn on their computer

especially to compose a new message—such as to invite people to a party—rather than use their at-hand mobile phone. Interviewees preferred the visual web interface to the console syntax because it was easier to use, offered more control and they felt less likely to make an error. New users found starting a new Rhub conversation using the console syntax difficult. Each attempt at sending a message using SMS has a financial cost and there is the possibility that an erroneous message might humiliate the sender if it is accidentally broadcast to the wrong group, for example. Several participants reported keeping old Rhub messages (which include syntax hints) as templates to make message composition easier. Thus, using the web was seen as a "safer" option, which is reflected in the proportionally high level of usage for initiating conversations (Figure 3).

Errors

Using a text-based syntax allows Rhub to be easily made accessible with alternative systems. While there are many usability disadvantages with an opaque command-based system, we found that our syntax (the "console") could be used with minimal errors by most users. The average error rate for users for their first 25 weeks of usage is 21%, where an error is counted as when an error message is returned to the user by the console (e.g. incorrect syntax, unknown group or individual recipient, etc.). Average error rates per user did not depreciably drop over time. We reason this is because most users only used the console for messaging, and because other commands were so infrequently used, difficulty was encountered when they were needed again.

We noticed differences in error rates between channels. SMS, which in Australia incurs a cost to send a message (with message receipt being free of charge), places an economic pressure on the person to produce correct syntax and have their command performed. Resultantly, 65% of usage attempts were abandoned after one erroneous command. With other channels, users have the opportunity to easily retry commands, leading to a much lower abandonment rate (IM 21%, web 38%). It is worth noting here that our analysis only identified syntax errors which led to an error message being returned to the user. Logically erroneous but syntactically correct messages (e.g. incorrect recipient) were therefore not counted as errors so our figures may underestimate actual errors. Based on our other observations and analysis, however, we believe that such errors were rare.

To help the user repair from errors, we try to provide contextual error responses, such as providing a list of group-related commands if it looks like they are trying to manipulate a group, or providing example syntax if they are missing a required part of a command. For example, if the user issues ">*&mygroup add*", the system would reply with a response indicating they should include who to add to the group, and to try ">*&mygroup add bob jane mary*". After

receiving an error response, the user could alter their command and resend it, which often leads to success, for example, 71% of erroneous IM sessions eventually succeed, while only 36% of erroneous SMS sessions eventually succeed as they are often prematurely abandoned. A different billing model may result in different usage characteristics, for example if all messages sent to Rhub were free, we would expect to see more attempts.

Learning the syntax

To assist mobile users who cannot access the web-based help system, we produced a three-sided card (Figure 4) with a list of commonly used commands with examples, which can be slipped into a wallet or purse. Cards were distributed by the authors and occasionally other Rhub members who requested batches of cards to give to people they invited. It was also available to download and print. We observed people using the cards on a number of occasions as a quick reference for infrequently used commands.

We also observed group members showing each other how to use the console. A previously sent message might be called up on a phone and shown to someone as an example of how to send a message, or some "early adopter" users eagerly demonstrated more advanced features to others.

Message overload

Participants in the largest group spoke of the volume of messages they were receiving (mostly via SMS and IM), as active periods sometimes produced over 20 messages per day. The average interviewee reported receiving 52 text messages per month prior to Rhub (s=49), so there is a considerably larger quantity of messages that have to be managed by the person. As one participant said, *"[My] phone holds 150 messages - it's always getting full after Rhub"*. Two interviewees described being woken by Rhub messages during their sleep, with one now putting his phone on "silent" overnight while it charges at his bedside. Participants who received few text messages before joining Rhub formed new associations with their text message tone, saying: *"when I hear my phone beep, I know it's Rhub"*. When the group was out socially, there would be a cacophony when another member not present would send the group a message, as everyone's mobile beeped in series.

Figure 4. Quick-reference fold out card

Several users suggested marking of outgoing messages as being "chat" or not, so they could set appropriate filters and not receive chat messages on their phone, yet still receive organization messages. We did experiment with this, whereby messages could be marked by including a tag within curly braces, such as: *"Go the blues! {chat}"*. Users could then set per-group filters to determine which message would be forwarded to them (all messages would still be available on the web however). This system was not used to any appreciable degree. Because of the dual requirement of senders having to tag messages and receivers having to define filters, it ultimately was not worth their bother. We later modified this system so instead of requiring tags to be within braces, filters could be set based on words appearing in the message, and a few people used this to filter messages from group Alpha. We also added other features to help users cope with incoming messages, such as the ability to set to a "silence" which disables all Rhub notifications for a specified number of hours or days.

Presence: Location and Status

As designers, we were aware of certain features that can be provided if a culture of presence-information setting developed amongst users. For example, users can be notified when their friends are nearby, or the value of a location could be gauged by how many friends frequented it. We have reservations with automatic detection and disclosure of location and status. We believe there is more meaning to being "at" a location than being physically located there. For example, waiting for a bus in front of a pub at 11am does not necessarily infer a drinking problem; merely that it is a handy bus stop. There is no reliable and accurate way to automatically capture and determine this notion of being "at" a location. As such, we decided on manual disclosure of presence, which permits greater user control at the cost of greater effort. A location that exists in the database can be referred to by its canonical name or alias; for locations that don't exist, free-form text can be used. Most users (71%) set locations that were already defined (by either themselves or others), 17% used arbitrary text, and 13% used the in-built "home" alias for their "home base" as set in their profile. Presence was mostly set to people's work location and popular nightspots.

The presence feature underwent significant design evolution as we attempted to increase its usage among participants, and thus increase the flow-on benefits of additional data available to the system. During interviews, participants agreed on the value of presence information to others and several of their suggestions, such as a prominent prompt for presence information when logging in, were implemented. Participants however indicated that there was little personal reward for the effort to set presence and as there were few friends using the feature, the group usage rewards did not eventuate either.

To counter this, we progressively increased the visibility of presence information, and the feature generally. Originally a console-only feature (since we anticipated it be mostly
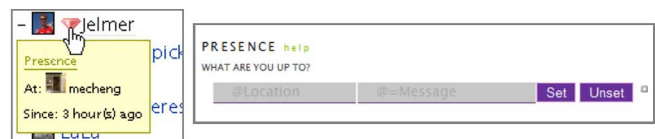

Figure 5. Icon indicates presence information is available (left); Presence prompt shown on index page after logging in (right)

used whilst mobile), we integrated the feature into the website more fully, allowing people to view others' presence history, see who's at a location when they are viewing it, and show icons next to user names when presence information is available (Figure 5).

We also introduced a feature that provided information on public transport. After setting a location, for example by sending "*@work*", the user could ask Rhub how to get to another location using a particular form of transport, or whichever is quickest, e.g. "*!bus home*" or "*!transport gym*". Rhub would reply with information on route numbers and times for the first three available departures in ten minutes' time. Users liked the transport feature and as a result, set their presence more often. However rather than setting their presence "properly" when they first arrived at a location, users were setting their location as they were departing the location, just to use the transport feature.

To further increase the visibility of presence setting, we introduced notifications for when a friend sets their presence. Generated notifications such as "*John is @ Tennis Club,*" are sent via SMS and IM to nearby friends or friends who've recently been to that location, and IM notifications are sent to remaining friends. Results of enabling notifications are preliminary; however, we observed a doubling of usage that has been sustained for three months so far.

By increasing the visibility of the presence feature using an iterative, user-centered design process we slowly evolved it to be more useful to the end-user. In a smaller scale study, we might have asked participants to set their presence regularly; but in our study we needed to appeal to users' self-interest. Groups of friends setting presence information might enable interesting features, however people naturally want to be able to see some selfish benefit before investing effort for a possible group benefit.

CONCLUDING REMARKS

In our discussion of a pervasive, multi-channel mobile social software system, we outlined observations that can inform the design of similar systems. Significantly, our results represent a large number of users over a long period of time, data which is lacking from most academic studies. By observing the system over a period of 18 months we saw how use patterns developed, repeated and diminished, and how the interface could be adapted and new features introduced through an iterative development method.

Upon reflection, we observe that the shift to facilitating group messaging as well as socialization across media engender specific kinds of use, even new kinds of use,

different from other social technologies. For example, our findings confirm other work showing that group messaging systems tend to be used for coordination more so than chat. One particularly common pattern of use that emerged was that of half-invites, where anyone within the group is invited to come along, often at short notice, without requiring any RSVP.

We found differences in the content and usage habits across channels, for example messages sent from a computer were more likely to refer to time and location than those sent using a phone, probably because fixed-location usage relates more to future activity where time and location must be defined, whereas mobile usage is "in-context", in the midst of activity.

Although the console syntax that allowed messages to be sent across media was not ideal, it was sufficiently usable that we observed long-term usage of the system by a number of social groups. A more intuitive interface could be developed, although the fact that usability was not a barrier for many people suggests an alternative. When paying attention to facilitating accessibility and utility it is worth making the system easily accessible and available on devices that people already own. In effect, designers must consider the broader parameters of what makes a system useful by considering the assemblage of devices and practices that are supported in the context of use rather than focusing solely on the interface or the design of a particular device.

ACKNOWLEDGMENTS

REFERENCES
1. Buur, J. and Soendergaard, A. Video card game: An augmented environment for user centred design discussions. In *Proc. DARE 2000*, ACM Press (2000).

2. Counts, S. Group-Based Mobile Messaging In Support Of The Social Side Of Leisure. In *Proc. CSCW'07*, ACM Press (2007), 75-97.

3. Farnham, S. and Keyani, P. Swarm: Hyper Awareness, Micro Coordination, and Smart Convergence through Mobile Group Text Messaging. In *Proc. HICSS'06*, ACM Press (2006), 1530-1605.

4. Gaver, B., Dunne, T. and Pacenti, E. Design: Cultural Probes. *Interactions* (1999). 21-29.

5. Grinter, R.E. and Eldridge, M.A. y do tngrs luv 2 txt msg? In *Proc. ECSCW'01*, Kluwer (2001), 219-238.

6. Hirsch, T. and Henry, J. TXTmob: text messaging for protest swarms. In *Proc. CHI'05*, ACM Press (2005), 1455-1458.

7. Höflich, J.R. and Gebhardt, J. Changing Cultures of Written Communication: Letter - E-mail - SMS. in Harper, R., Palen, L. and Taylor, A. eds. *The Inside Text* Springer, Dordrecht, The Netherlands, 2005, 9-31.

8. Jung, Y., Persson, P. and Blom, J. DeDe: Design and Evaluation of a Context-Enhanced Mobile Messaging System. In *Proc. CHI'05*, ACM Press (2005).

9. Kopomaa, T. The Breakthrough of Text Messaging in Finland. in Harper, R., Palen, L. and Taylor, A. eds. *The Inside Text*, Springer, Dordrecht, The Netherlands, 2005, 147-159.

10. Korteum, G. and Segall, Z. Wearable Communities: Augmenting Social Networks with Wearable Computers. *Pervasive Computing*, 2:1 (2003). 71-78.

11. Ling, R. *The Mobile Connection: The Cell Phone's Impact on Society*. Mogan Kaufmann Publishers, San Francisco, CA, USA, 2004.

12. Ling, R. The Sociolinguistics of SMS: An Analysis of SMS Use by a Random Sample of Norwegians. in Ling, R. and Pedersen, P.E. eds. *Mobile Communications*, Springer, London, 2005, 335-349.

13. Ling, R. and Yttri, B. Hyper-coordination via mobile phones in Norway. in Katz, J. and Aakhus, M. eds. *Perpetual Contact*, Cambridge University Press, Cambridge, 2002.

14. Maden, A., Caneel, R. and Pentland, A. GroupMedia - Using Wearable Devices to Understand Social Context. In *Proc. MobiSys'04*, ACM Press (2004).

15. Matsuda, M. Mobile Communication and Selective Sociality. in Ito, M., Okabe, D. and Matsuda, M. eds. *Personal, Portable, Pedestrian*, MIT Press, Cambridge, MA, USA, 2005, 123-142.

16. Okada, T. Youth Culture and the Shaping of Japanese Mobile Media: Personalization and the *Keitai* Internet as Multimedia. in Ito, M., Okabe, D. and Matsuda, M. eds. *Personal, Portable, Pedestrian*, MIT Press, Cambridge, MA, USA, 2005, 41-60.

17. Schusteritsch, R., Rao, S. and Rodden, K. Mobile Search with Text Messages: Designing the User Experience for Google SMS. In *Proc. CHI'05*, ACM Press (2005), 1777-1780.

18. Sillence, E. and Baber, C. Integrated digital communities: combining web-based interaction with text messaging to develop a system for encouraging group communication and competition. *Interacting with Computers*, 16 (2004). 93-113.

19. Smith, I., Consolvo, S., Lamarca, A., Hightower, J., Scott, J., Sohn, T., Hughes, J., Iachello, G. and Abowd, G.D. Social Disclosure of Place: From Location Technology to Communication Practices. In *Proc. Pervasive'05*, Springer-Verlag (2005), 134-151.

20. Weilenmann, A. Negotiating Use: Making Sense of Mobile Technology. *Pers. Ubiq. Computing*, 5:2 (2001). 137-145.