Expressive Messaging on Mobile Platforms

Per Persson, Jussi Karlgren, Panu Korhonen, Janet Galore, Mark Tierney, Chad Redmon, Juha Hemánus, Peter Lönnqvist, Jarmo Laaksolahti

April 25, 2001

Swedish Institute of Computer Science

Box 1263, S-164 29 KISTA, SWEDEN

mobilepal@sics.se

ABSTRACT

We present a design for expressive multimodal messaging on mobile platforms. Strong context, simple text messages, and crude animations combine well to produce surprisingly expressive results.

Keywords

Expressive messaging, Mobile devices, SMS, Multimodality

Expressive Messaging on Mobile Platforms

Per Persson[1], Jussi Karlgren[1], Panu Korhonen[2], Janet Galore[3], Mark Tierney[4], Chad Redmon[3], Juha Hemánus[2], Peter Lönnqvist[1,5], Jarmo Laaksolahti[1]

1) HUMLE laboratory, SICS, Box 1263, 164 29 Kista, Sweden

2) Nokia Research Center, Helsinki, Finland

3) Honkworm, Seattle, U.S.

4) room33, Stockholm, Sweden

5) Currently at Department of Computer and Systems Sciences, Stockholm University/KTH, Sweden

mobilepal@sics.se

ABSTRACT
We present a design for expressive multimodal messaging on mobile platforms. Strong context, simple text messages, and crude animations combine well to produce surprisingly expressive results.

Keywords
Expressive messaging, Mobile devices, SMS, Multimodality

CURRENT PRACTICE IN MOBILE MESSAGING
In certain parts of the world, mobile phones are everywhere and used by everyone. Phones are not only used for synchronous and asynchronous voice communication, but also – increasingly – for written communication. In spite of complex and non-intuitive interfaces, Short Message Service (SMS) messages are sent and received by a huge number of customers, mostly young, in many European countries. In Finland, the mobile phone users sent in total 650 million text messages in 1999 (approximately 100 SMSs/citizen annually) [1].

SMS differs from voice communication in many respects. First, it is not primarily used for serious and task-oriented communication, but for expressive, social and emotional functions, e.g. 'how are you doing?' and 'whazup?' messages. Humor, flirts, gags and play are central objectives of textual messaging. In spite of the low bandwidth (up to 160 characters, typically in low-resolution monochrome), there is a surprisingly high degree of expressivity in SMS communication, due to the systematic reliance on a rich and shared awareness of situation, preferences, sense of humor, and social context between sender and recipient. SMS messages are typically not sent to strangers, but used in peer-to-peer communication between friends, lovers, or family members. In addition, the composition and reception of SMS messages often take place in a collaborative setting with a group of people gathering around the device [2].

In the *Expressive Messaging* project, we want to maintain the special features of SMS messaging, but expand it into other areas of interest. The basic research and design questions are: What is expressivity? How it can be enhanced?

Avatars
Many collaborative multi-user environments allow people to use an *avatar* – a virtual representation of personal presence and personal characteristics. The way people choose to represent themselves in a virtual world is not always straightforwardly predictable from their physical world appearance and character – the virtual world allows users to play with their identities [3]. Our design project allows users to employ animated characters to enhance expressivity of SMS messaging. However, the *synchronous* nature of most virtual avatars makes fine-grained expressiveness difficult since users will be unable to control the behavior of the avatar online over a simple numeric keyboard. In this project, we explore *asynchronous* avatar usage.

Emoticons
In this respect, our project draws more from *emoticons* than from avatars. Combining a textual message with a semi-imagery representation of a face adds new layers of meaning (e.g. irony), guides the recipients' interpretation of the message, and expresses the sender's emotional state. Creating animated emoticons that move and transform over the temporal flow of the message will merge modalities in similar ways to cartoon and animated film. The efficiency of emoticons also shows that expressivity lies not in the realism of the imagery, but in the combination of modalities, and, again, the rich and shared context of sender and recipient.

DESIGN CHOICES

Composing Messages: Moods and Events

The first design issue was how to allow senders to express a basic emotional undertone in the message through the actions of their avatar. Users start composing a message by selecting from a palette of *moods*, e.g., happy, distressed, angry and busy. This makes the avatar perform some action to reflect the mood in a few seconds of animation, which will then be looped during the length of the message. In parallel, the textual message is shown as a cartoon balloon.

In addition to mood animations, avatars should be able to perform simple *events*, e.g. jump, dance, smile, laugh, weep (possibly connected to sound effects). Events can be added at specific points in the message. In preview mode, the animation can be stopped between the looped mood animations and the user can choose between events in a list. Since all mood and event animations start and stop at a neutral position frame, continuity of movement can be ensured without involving the user. When the sender has inserted event(s) and previewed, the package is sent off.

Events need to be paced to match the written message – timing is central in conveying punch lines or other emotional content. To this end, we have sketched a standard for encoding expressively enhanced messages: the Mood Markup Language or *MoodML* (Figure 1).

Figure 1. MoodML example.

```
<MoodML skin="88F2A0F" id="janet"
 to="0405047123" mood="happy;8">
I'm excited to see you!
<event="jump">
Be there at ten.
<event="dance">
</MoodML>
```

Quickies and Recycled Messages

In some cases, standard scripted responses to messages can be useful. Our design gives recipients the possibility of saving the script from a received message and recycling it later, using his or her own character skin and rewriting the balloon text.

Skins

Expressivity lies, however, not only in the temporal composition of text, moods and events, but also in the graphical design of the *skins* and movements. Most users will probably make use of prefabricated skins. Some standard skins may be included in the service package. Others may be produced by professional firms for business marketing: just like product placement in films, users can get professional skins for free, if she allows her mobile avatar to wear a *Budweiser* T-shirt in all her messages.

Some users, however, will want to design their own skins using their own graphical software. To this end, open standards and APIs are absolutely essential. No single organization will have the stamina to produce and uphold the interest of the potential user community: the creativity of the users must be tapped into. Thus, it is important to create a community in which sharing scripts and skins are encouraged and awarded.

However, in order to ensure compatibility between skins, scripts, and users, there have to be some minimal requirements for any given skin. For instance, if you make a skin publicly available or start using it yourself, it has to perform a minimum list of mood and events.

Some users will want to switch between different skins depending on context, e.g., messaging with buddies, colleagues, relatives, and partners. The system provides easy configuration possibilities for this.

Configuration

Configuring the interface and the characteristics of the skin, archiving message scripts, sharing scripts and skins, community building and other administrative aspects of the system are accessed by the user from a Web interface rather than over the mobile device: the mobile device is used solely for composing, transmission and receipt of messages. We expect most users will not be concerned with the niceties of configuration; but we do expect some users to be very interested.

Figure 2. Example screen.

IMPLICATIONS FOR HCI

Composing a message on a mobile device is different than doing it on a stationary terminal: using "dead" time on buses, in bars, and on school hour breaks will give ample time to compose and perfect avatar-based messages. Such non-work related usage of information technology has been an underdeveloped area of research in HCI.

Future research directions include exploring the tradeoff between reality and real-time concerns – familiar from virtual reality and natural language processing projects – and the exploration of autonomous or semi-autonomous character agents with scripts to constrain their behavior.

REFERENCES

1. Ministry of Transport and Communications, Finland. "Suomen Tekstiviestimarkkinat", Edita, Helsinki, 1999

2. Weilenmann, A. (2000) Negotiating Use: Making Sense of Mobile Technology, *The Journal of Personal Technologies*, no. 4, Springer Verlag.

3. Turkle, S. (1995) *Life on the Screen*. New York: Touchstone.

Instructions for Operating the Interactive Illustration *Expressive Messaging* (*MobiPal*)

Start-up
Open the submitted file named mobipaldemo.exe or mobipaldemo.swf.

Index Page
From this index, you can reach the four different scenarios in the demo, and exiting any of the four will bring you back to this page. Pressing Exit in the right hand corner will quit the demonstrator.
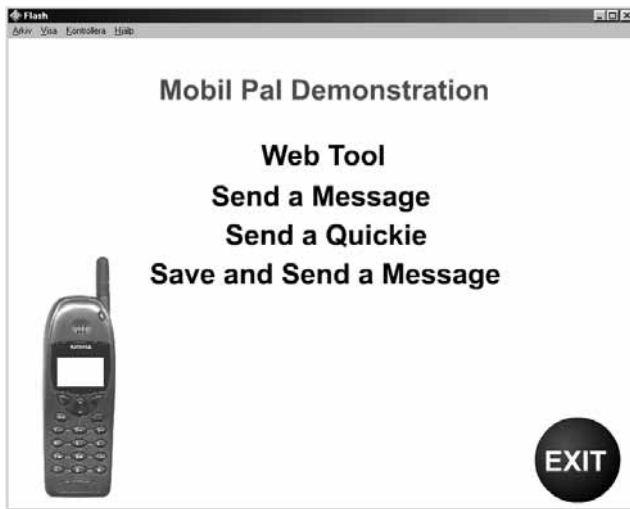

The Web Tool
This is the mock-up of the web configuration tool. In the actual system, this will be the place from which the user will be able to configure and adopt the information space presented in the mobile device. Many of the buttons in the illustration are not responsive.
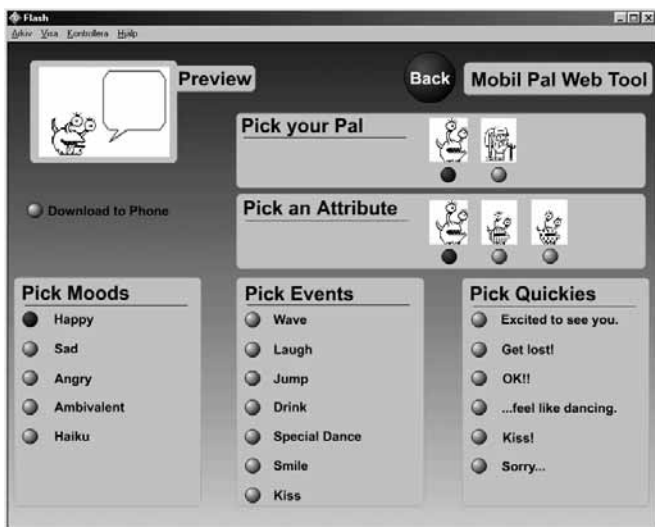
1. First, 'Pick your Pal'. This is the skin of the user's avatar. **Choose the 'blob figure'.** The figure will appear in the Preview window, which shows all the selections made by the user (immediate feed-back).

2. In the 'Pick an attribute' window, the user adds some attribute to the skin (color, texture) or possible some other attribute not present in the demonstrator (e.g. a rose or a sword). **Choose the neutral skin color**.

3. In the 'Mood window' the user chooses the identity and number of the basic moods the avatar can have. This will provide the basis for expressive messaging later in the demo. **Choose Happy** and the embodied representation of 'happy' will appear in the preview window.

4. Next, the user chooses a couple of (punctual) events, which can be inserted at different points in the expressive message (se below). Moods consists of 4 seconds animation loops and in-between those loops, events may be inserted. **Choose jump and dance** and make sure those choices can be previewed properly in the preview window.

5. Finally, quickies are pre-scripted sequences of events, which have been stored by the user for quick responses to expressive messages from other people. The demonstrator does not allow any choice to be made here.

6. Finally, **press 'Download to Phone'** which will bring you into the second scenario 'Send a Message'.

Send a Message
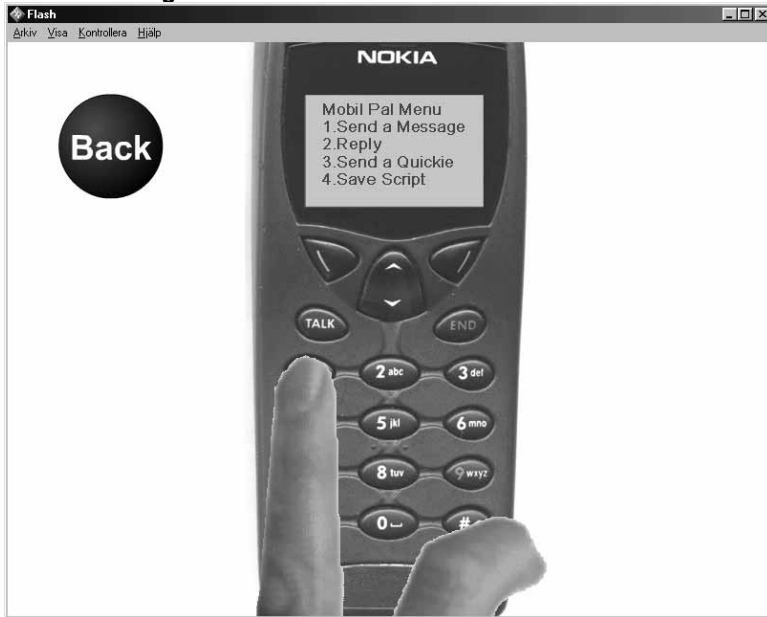
This scenario takes you through the steps in sending a basic message. Via the mouse/finger, you will be able to interact with the visual representation of the phone.

1. From the Main menu, **choose 'Send a Message'** (by pressing the '1' key on the phone).

2. 'Pick a Recipient' – **choose 'Betty' (2)**.

3. 'Pick a mood' – **choose 'happy' (2)**.

4. 'Pick length'. By pressing the top, middle button you will be able to increase the length from 4 to 8 and finally to 12 seconds (in practice this will mean 1,2 or 3 four seconds loops). **Choose 12 seconds and press the 'talk' button**.

5. This will bring you to the 'enter message' phase in which the demo automatically simulates a user entering a message ('I am excited to see you. Be there at ten').

6. After the message is inserted, **press 'Talk'.** Automatically, the composition will be reviewed (a 12 second message with text inserted in the written order). The review will automatically loop when finished.

7. Now, our hypothetical user will want to insert a punctual event at some points in this message: she wants to add a jump after the text 'I am excited to see you', and a dance at the end of the message, after the text 'Be there at ten'. In order to do so, review the sequence, and at the appropriate point in time (right after 'I am excited to see you'), **press 'Talk'** which will bring up a menu of possible events. **Choose 'jump'** (1) which will bring you right back to review mode again. This time the sequence will consist of the happy mood and text, but with the jump inserted after the text 'I am excited to see you'.

8. In order to insert the dance at the end of the message, do the same steps as (7): After the text 'Be there at ten', **press 'Talk', and choose 'Dance' from the menu**. Next, the final sequence will be reviewed (now with the sequence happy-jump-happy- dance).

9. When the animation comes to an end, **press 'End' to ok.**

10. 'Sending message?' – **press 'Talk'** and the message will be sent.

11. To go back to the main index, **press exit**. (If you want to repeat only some of the steps in this scenario, you will, unfortunately, have to exit here and go through all of the steps again)

Send a Quickie

This scenario demonstrates how a hypothetical recipient (Betty) receives and responds with quickie to a message sent by her friend Janet (the message composed in the 'Send a message' scenario above).

1. First, Betty receives Janet's composition ('I am excited to see you. Be there at ten' in a sequence of happy-jump-happy-dance).

2. After the message, the MobiPal Menu will pop-up. **Choose 'Send a Quickie'**.

3. 'Choose a recipient' – **choose 'Janet'**.

4. 'Pick Quickie' – **choose 'shout' (1)**.

5. The quickie is reviewed by Betty.

6. **Press 'End' to send.**

7. **Press 'Talk' to send.**

8. To go back to index page, **press exit.**

Save and Send a Message

In this scenario, a hypothetical user (Betty) is receiving a sequenced message from Janet. Betty likes this particular sequence of events (happy-jump-happy-dance) and saves this sequence script. Then, she uses this script to compose a message in her own skin and with a new text.

1. Betty receives the message from Janet.

2. The MobiPal menu appears – **choose 'save script' (4)**.

3. The system confirms that the script has been saved.

4. **Press 'Talk'** and the main menu appears again. Now Betty wants to reuse the script in her own skin. **Choose 'send message' and choose 'Janet' as recipient.**

5. 'Pick message type' – **choose 'Script'**.

6. 'Pick script' – **choose 'I am excited'** (which is the script Betty just saved).

7. 'New Message' (or better: text) – **choose yes ('Talk').**

8. Betty enters a new text for the message : 'Love to see you'.

9. **Press 'Talk'** and the composition will be reviewed. Please note that the sequence of events is the same as Janet's (happy-jump-happy-dance), but that the skin is Betty's which will bring another expressivity to the sequence. In addition, Betty has used another text to fit more with the objectives of the message.

10. **Press 'End'.**

11. To send choose 'yes' ('Talk').

12. The system confirms that message is being sent.

13. To return to the index, press exit.