December 2006

Enhancing Asynchronous User Communication with Cross Platform and Channel Agnostic Messaging Services

Jan Muntermann
Chair of Mobile Commerce & Multilateral Security- Goethe University

Natasa Milic
Integrated Systems Microsoft Research Ltd

Recommended Citation

Muntermann, Jan and Milic, Natasa, "Enhancing Asynchronous User Communication with Cross Platform and Channel Agnostic Messaging Services" (2006). *AMCIS 2006 Proceedings*. 252.
http://aisel.aisnet.org/amcis2006/252

Enhancing Asynchronous User Communication with Cross Platform and Channel Agnostic Messaging Services

Jan Muntermann
Chair of Mobile Commerce
& Multilateral Security, Goethe University
Frankfurt, Germany
jan@whatismobile.de

Natasa Milic Frayling
Integrated Systems
Microsoft Research Ltd
Cambridge, United Kingdom
natasamf@microsoft.com

ABSTRACT

Choosing an optimal channel for communication has become a difficult task for the user. The number of options continuously increases with new devices and services. For asynchronous communication we nowadays use several mobile messaging services such as Short Message Service (SMS) and Multimedia Message Service (MMS), in addition to ubiquitous e-mail that is available on the desktop and mobile devices. Users face two main difficulties. First, the sender has no information about the recipient's device status or preferences. Second, the sender may not have access to a messaging client or online messaging service that supports message types the recipient can receive. These challenges represent opportunities for new services that we explore through design science research. We develop a system architecture and client applications for communication across Internet and mobile cellular networks. As the user composes the content of a message, the system recommends the optimal service. Based on the sender's selection, it constructs the appropriate message type and sends it to the recipient. The paper describes the system architecture and the recommender model and assesses the properties of the system in relation to the system requirements.

Keywords

Mobile messaging, media choice, design science, IT artifacts

INTRODUCTION

We are witnessing a proliferation of messaging services that use different infrastructures and technologies, such as Internet based e-mail, instant messaging services, Short Messaging Service (SMS), and similar. Their availability offers flexibility but, at the same time, introduces new complexities. It is often difficult to decide what means of communication to use in a particular instance to communicate optimally. There are many factors that influence such a decision, from social to economical. Objective of our research is to explore the ways we can assist the users in making the right choice. For the sake of concreteness we focus on technical aspects of a system that facilitates asynchronous communication via services such as e-mail, SMS, and MMS.

Typically, the sender of a message is faced with several decisions, with little information to rely upon:

What is the most appropriate messaging service to use, having in mind the specific characteristics of the message content?

What is the best way to deliver a message to the recipient, having in mind the recipient's connectivity status and available communication services?

How can the two prior goals be attained while optimizing the total utility for the sender and the recipient, including the communication costs?

Furthermore, even if the user can identify the most appropriate messaging service, the user may not have access to the client application that supports the particular message type. Generally, both the sender and the recipient need to have access to the same type of communication channel in order to communicate. MSN Messenger and ICQ provide a limited message conversion that enables users to send SMS messages through the Instant Messaging client. Some mobile phone services provide an e-mail gateway that enables the users to send a text message from an e-mail account to an SMS subscriber. However, the need for cross-protocol communication that integrates diverse communication channels is not fully addressed.

In summary, users face new challenges due to the increasing number of message services available, the coexistence of different and service dependant user accounts, and the lack of mediation between message senders and recipients. In this paper we define functional requirements of a client-server system that aims to address some of these issues. We built a prototype to assess the feasibility of implementing such a system. Through methods of the design science research we evaluate the current implementation against the functional requirements and provide recommendations for the next phase, aimed at a system that is suitable for deployment and user evaluation.

PROBLEM DEFINITION AND LITERATURE REVIEW

Given the described problem areas, we identify two important issues.

First, due to the proprietary message formats, cross-application and cross-platform communication is not well supported. The roaming capabilities of messaging services across GSM and IP networks, for example, are limited. Support for effective network roaming requires conversion of message types and implementation of the appropriate network bridging layer.

Second, faced with several options for sending a message, the users need help with optimizing their communication objectives while respecting the recipients' preferences. Currently this is rather difficult since the sender typically has no information about the recipient's reachability.

In Table 1 we provide functional requirements for a system that would address these issues.

Functional Requirements		Description
(1)	Universal messaging client	While creating a message, the sender may have limited information about available communication services or may be undecided about the message type to use. Therefore, a client for generic message editing is required.
(2)	Message roaming across different network types	Messaging across different networks requires infrastructure for roaming capabilities, e.g., across GSM and IP networks.
(3)	Conversion of message types; aim towards standard message formats	A generic message composed by the sender needs to be converted into the format of the user selected message type.
(4)	Decision support: recommendation for selecting an appropriate messaging service for a given message	As the number of available communication channels increases and the content types diversify, the sender needs support for selecting the most suitable communication channel.
(5)	Capture and incorporate information regarding recipient connectivity	A complete lack of information about the recipient can lead to a selection of an unsuitable communication channel and thus delay the message delivery or increase the messaging costs.

Table 1. Channel Agnostic Messaging: Problem Summary and Functional Requirements

Related Research

Prior research has focused on two ways of supporting user communication: by increasing the scope of information available to the sender and by providing a simple message conversion. Maniatis, Roussopoulos, Swierk, Lai, Appenzeller, Zhao and Baker (1999) designed a prototypical infrastructure called Mobile People Architecture that supports routing of incoming messages to multiple user accounts. They increased person-to-person's reachability by mapping users' names to application-specific addresses and automatic message conversion. However, they did not address the loss of content that occurs when some message parts cannot be converted. Consequently, the system is prone to message impairability without notification of either the sender or the recipient.

Tang, Yankelovich, Begole, Van Kleek, Li and Bhalodia, (2001) propose to increase the user awareness about other individuals and their connectivity status. In order to assist the sender in choosing the communication service, the system shows all currently available communication channels and highlights those that were active most recently. However, it does not inform whether that channel is appropriate and compatible with all message elements. Furthermore, it suffers from the privacy and security problems since user sensitive information is exposed. Finally, it requires manual browsing through all communication channels available to the recipient while some of them are not compatible with the clients or services the sender is using.

Nakanishi, Kumazawa, Tsuji and Hakozaki (2003) describe a mobile communication tool that supports communication within closed user groups. Their design requires location and schedule information to be shared among the members. This raises privacy and security issues, especially for wider user groups. Furthermore, message conversion is not supported.

Moon and Sanders (2004) conducted a user study to investigate perceived usefulness of different communication media and discuss how they vary depending on the user's objectives. King and Xia (1997), on the other hand, researched factors that influence adoption of new communication media. They found that the individual user's learning experience and training are essential for both the media choice and perceived media appropriateness. These studies provide useful insights in the social aspects of communication. Their techniques and findings will inform the prototype deployment and user evaluation we plan to perform in the subsequent phase of our work.

Design Science Research Methodology

The distinction between the natural science and the design science, and its implications for Information Systems (IS) research is a topic of ongoing discussion (Simon, 1996; March and Smith, 1995; Hevner, March and Park, 2004). While natural science is based on discovering and justifying scientific concepts, with the aim of understanding and explaining observations, design science focuses on the creation and evaluation of useful artifacts in order to provide value for users (March and Smith, 1995). In the IS literature, the importance of design science has been well recognized as the design of useful and innovative artifacts can provide new knowledge where the existing theory meets its limits (Hevner et al., 2004).

In recent years, several authors applied design research science methodology building IT artifacts in different IS application areas. Markus, Majchrzak, and Gasser (2002) designed and deployed a software tool TOP Modeler for developing information systems that support processes for semi-structured and unstructured decision making. They derived a design theory that supports developers in specifying software feature lists. Tiwana and Ramesh (2001) designed a prototypical application to support teams that produce information products. They created a knowledge management system that supports collaborative decisions. It combines distributed authoring with synchronous deliberations that can be captured and linked to different versions of the information product.

However, the IS community has also been criticized for relatively limited use of design science research, considering the benefits the exploration of IT artifact can have for developing new theories in IS research (Orlikowski and Lacono, 2001; Au, 2001; Srinivasan and March and Saunders, 2005). Since existing theories often cannot adequately explain observations, we need methods to reach beyond their limits. Markus et al. (1992) point out that the descriptive and evaluative nature of IS design research is not a shift from the established IS research but rather its extension that enables us to introduce and explore new ideas, concepts, and designs, and stimulate IS research in general.

March and Smith (1995) identified four types of artifacts that originated from the design science research in IS. These are constructs, models, methods, and instantiations that are first deployed and then evaluated for their utility or ability to solve problems in a given application domain. Srinivasan et al. (2005) use a more specific definition of IT artifacts as hardware and software systems that are constructed for a specific purpose. The IT artifacts are designed, deployed, and evaluated for appropriateness and effectiveness within a particular organizational context.

Hevner et al. (2004) introduce a framework for conducting design science research in the IS field. They define several research guidelines which inspire and guide our work. First, the result of design science must be an IT artifact addressing an important business problem within a relevant domain. Second, the artifact should be evaluated with an appropriate assessment method in order to identify relevant research contributions. Third, both the artifact deployment and the evaluation should be conducted in a rigorous and iterative manner, aimed at achieving a superior artifact through adequate modifications. Finally, the results need to be presented appropriately for the technology and management oriented community. By designing, implementing, and evaluating IT artifacts it is possible to analyze and understand the problem space, assess how feasible the artifact is as a possible solution, and how effective the design is in enabling researchers to evaluate the utility the artifact provides (Nunamaker, Chen and Purdin, 1991; Hevner et al., 2004).

In this paper we describe IT artifacts, in particular the system infrastructure and software prototypes that we designed and implemented to investigate support for user-to-user channel-agnostic messaging. We adopt the iterative design and implementation approach. At this stage we evaluate properties of the implemented prototypes in relation to the functional requirements and provide recommendations. We intend to use these findings as a starting point for the second round of design and implementation, aimed at the system deployment and user evaluation. The ultimate goal is to devise a system that can be made available for a broad consumption.

INFRASTRUCTURE SETUP AND BACK-END SERVICES

We designed and implemented a client-server system called mCommunicator that supports a seamless usage of different communication devices and messaging technologies. Integration is achieved by the use of Web Services. The infrastructure is accessible via wireless (e.g., GPRS) or conventional (e.g., Ethernet) Internet connections and therefore supports both the mobile and the desktop connectivity. Figure 1 depicts the mCommunicator client-server architecture.

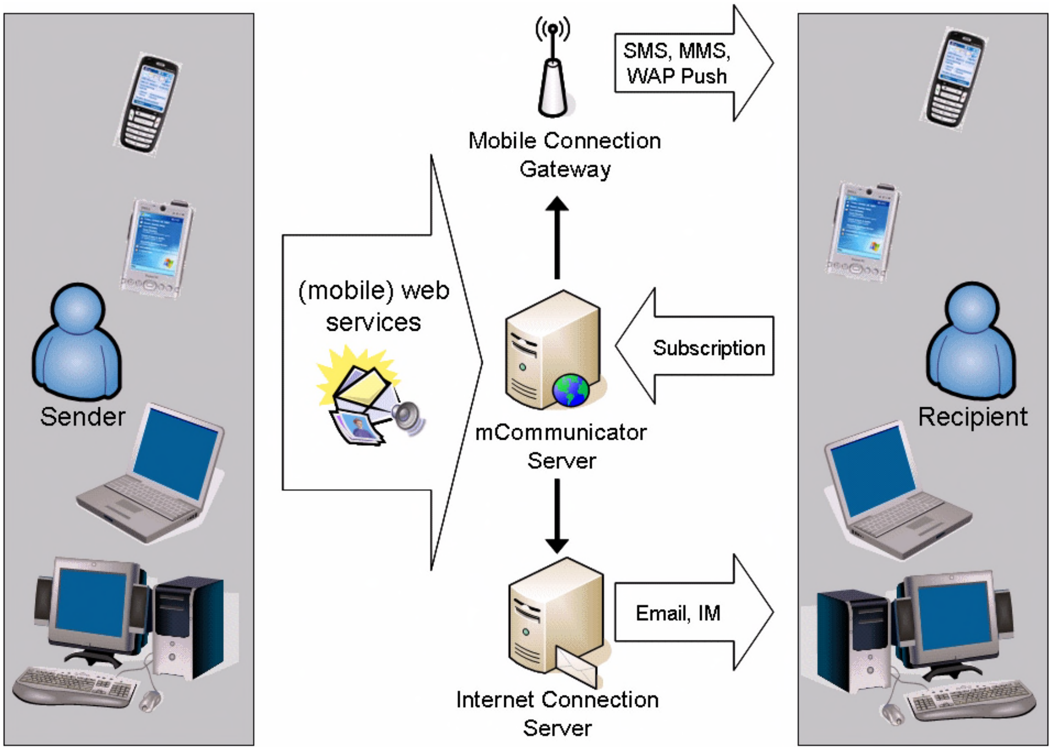


Figure 1. Infrastructure Setup Overview

The mCommunicator server is responsible for: (1) managing the users' connectivity status, including information about the available communication channels, (2) storing, 'packaging', and converting the content of a message into a required format, (3) roaming messages across different networks, and (4) recommending appropriate communication channels to the sender. The mCommunicator infrastructure and service components are aimed at meeting functional requirements outlined in Table 1. We describe its architecture and features in more detail in the following sections.

User Status and Message Recompilation

We use an SQL database to store and continuously update information about the user, including the user's preferences, messaging service subscriptions, connectivity status, and the content of the user's messages. The user's connectivity status is obtained directly from the user's devices. As the user turns on a device, the device registers with the mCommunicator service, indicating that it is available for communication. This is further supplemented by the user's login. The connectivity information is not directly revealed to the community of users. Instead, it is used by the recommender engine as one of the parameters when providing advice to the sender.

Message components and attributes are obtained from the message composed by the sender. As the sender notifies the mCommunicator server of the message content, all message components and attributes are stored in the database. When the selection of the communication channels is completed, they are used to recompile the message in the required format.

Message Roaming with Messaging Gateways

mCommunicator service comprises gateways that enable delivery of messages across communication services and networks. The current prototype supports message processing and delivery through e-mail services and asynchronous communication via GSM networks. Within a GSM network it supports Short Message Service (SMS), Multimedia Message Service (MMS), WAP Push Service Indication (WAPSi), and WAP Push Service Load (WAPSl). The infrastructure is expandable and can be augmenting with additional messaging gateways, e.g., for instant messaging.

Communication Channel Recommendation

We propose a model of the communication space that enables us to quantify the utility of a particular asynchronous messaging service given a message that needs to be sent to the recipient. We apply the model to person-to-person communication where a sender specifies the content and attributes of a message and the system responds with recommendations that meet the delivery constraints and user preferences (see Figure 2). The model can also be applied to a wider context of personalized information delivery by services to individuals.

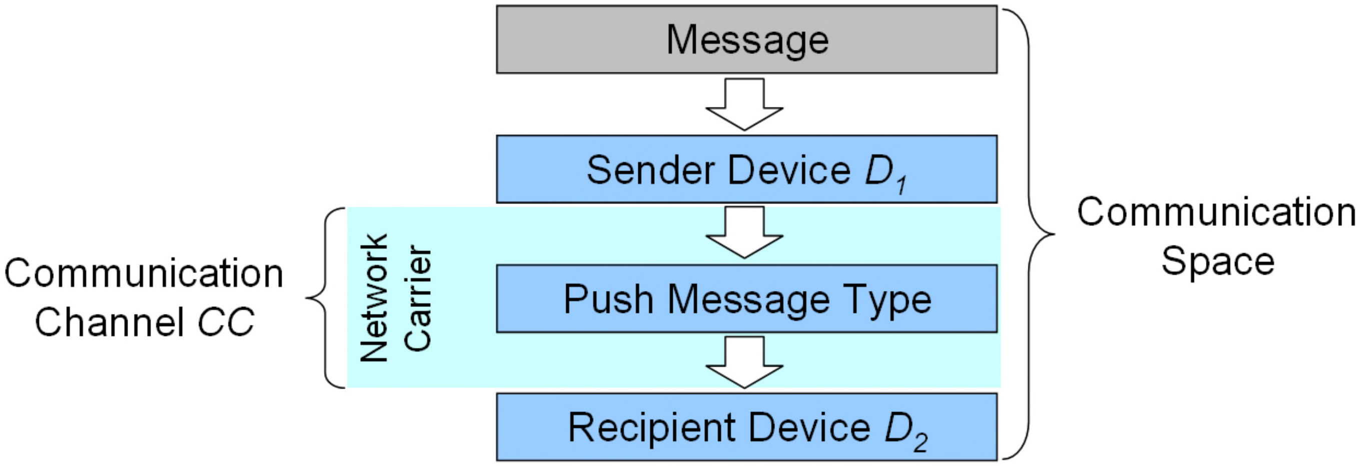


Figure 2. Communication Space Model Overview

Our representation of the model components includes vectors of attributes for message content, sender's and recipient's preferences, device properties, and communication channels via which the messages are to be sent. The attribute vector $A = (a_1, a_2, \ldots, a_n)$ describes the entire set of relevant attributes for the communication space. Message characteristics are captured by a function M over the attribute value space A, $M{:}A \ddagger M$ with $A \wp R^n$ and $M \wp R^n$, where n is the number of components in the vector and R is the set of real numbers. Each component function $m_i : (a_{1,}, \ldots, a_n) \ddagger m_i(a_1, \ldots, a_n)$ represents one characteristic of the message to be delivered. The importance of individual message attributes is described by a weighting vector $w = (w(m_1), w(m_2), \ldots, w(m_n)), w(m_i)$Œ[0,1]. Typically, $w(m_i)$ defines a threshold (in percentage of m_i value) required for the system to take the particular message characteristic into account. Otherwise the attribute is disregarded. Accordingly, $w(m_i)$ with zero value suggests that the attribute m_i is not required for the particular message.

A communication channel is a combination of the network connection (e.g., GPRS) and the message type (e.g., MMS). It is described as a vector CC with components $cc_i : (a_{1,}, \ldots, a_n) \ddagger cc_i(a_{1,}, \ldots, a_n)$ ŒR, $i=1,\ldots k$. This attribute vector shows the characteristics and functionalities of the given communication channel. The compatibility requirement implies the constraint $m_i \wp cc_i(A)$; otherwise a particular channel cannot support the chosen message type.

Accordingly, the device characteristic vectors d_j, j =1,2 describe features of the sending and receiving devices D_1 and D_2, respectively. Their attributes are defined as $d_{j,i}$: $(a_{1,}, \ldots, a_n) \ddagger d_{j,i}(a_{1,}, \ldots, a_n)$ ŒR, j =1,2 with $d_{j,i} \wp cc_i(A), i=1,\ldots k,$ i.e., attribute values are in the range of the corresponding channel attributes. Otherwise the device is not supported by the given infrastructure.

The utility of sending a message via a communication channel CC is calculated for each attribute i, $i=1,\ldots, n$:

$$U_{CC,i} = \begin{cases} 1 & if\ cc_i\ \ddot{Y}\, d_{1,i}\ \ddot{Y}\, d_{2,i} \quad m_i \\ 0 & if\ cc_i /\ d_{1,i} /\ d_{2,i} < m_i - m_i \quad w(m_i) \\ 0 & if\ m_i \quad w(m_i) = 0 \\ Min\left(cc_i/m_i ; d_{1,i}/m_i ; d_{2,i}/m_i\right) & else. \end{cases}$$

The communication costs associated with the communication channel *CC* are also defined as a function of the general attribute vector:

$$C_{cc} : (a_1, a_2, ..., a_n) \mapsto C_{cc}(a_1, a_2, ..., a_n)\,\text{Œ}R\,.$$

The cost C_{cc} is the total cost incurred by sending the message via *CC*. For convenience we also introduce the *utility relation vector* U^R_{PQ} that captures the relative utility of sending a message via communication channel *P* verse channel *Q*.

$$U^R_{PQ,i} = \begin{cases} U_{P,i}/U_{Q,i} - 1 & if\ U_{P,i}/U_{Q,i} \quad 1 \\ -U_{Q,i}/U_{P,i} + 1 & else, \end{cases}$$

$$for\ i = 1 \ldots n.$$

Similarly we use the *cost relation* C^R_{PQ} to compare costs of two communication channels *P* and *Q*:

$$C^R_{PQ} = \begin{cases} C_Q/C_P - 1 & if\ C_P/C_Q \quad 1 \\ -C_P/C_Q + 1 & else. \end{cases}$$

We concatenate the normalized and dimensionless cost and utility vectors into the *relation vector* R_{PQ}

$$R_{PQ} = concat\left(U^R_{PQ}\ C^R_{PQ}\right).$$

This simplifies the computation of comparative measures that determine whether a communication channel *P* is more adequate than *Q*. A comparative measure *f* can, for example, be defined as a simple function of vector components, such as the weighted mean, median, or a more sophisticated estimation functions. By using a weighted average, it is possible to define the relevance for each of the elements in the relation vector R_{PQ}. For that reason we determine the preference for channel *P* compared to a channel *Q* as:

$$f_{P,Q} = \frac{\mathop{\Re}\limits_{i=1}^{n} w_i \quad U_{PQ,i} + \mathop{\Re}\limits_{i=1}^{n} w_i \quad C_{PQ,i}}{\mathop{\Re}\limits_{i=1}^{n} w_i}.$$

The resulting two-dimensional matrix is used to identify the most appropriate among the available communication channel. Table 2 illustrates a typical set of attributes for communication channels. In Table 3 we show cost functions for different (mobile) devices in use.

Expanding the infrastructure with new communication channels or devices requires definition of the corresponding attributes, cost functions, and utility functions. When multiple devices and communication channels are involved, most of the work is associated with modeling individual devices. However, many devices can be handled as device classes as they share the operation system (such as Windows Mobile, Series 60 or Palm OS) and the respective messaging client.

When defining the total cost function we currently neglect the costs of roaming for the sake of simplicity. However, we recognize that the cost of roaming is important, particularly when the users who are connected to foreign mobile networks. The proposed communication space and the channel ranking model provide the basis for assessing the suitability of the mCommunicator service and the particular recommender system.

	Communication Channel *CC*				
	SMS (GSM)	MMS (GPRS)	WAP Push SI (GSM)	WAP Push SL (GSM)	Email (Ethernet)
Creation Date	0	0	1	0	1
Priority levels	0	1	1	1	1
Link to external services	0	0	1	1	0
Delivery confirmation	0	1	1	0	1
Read Confirmation	0	1	0	0	1
Sender identification	0	0	1	0	1
Message class attributes	0	1	0	0	0
Message Subject Field	0	50	0	0	255
(Maximum) message size	160*	U(d)	U(d)	0	U
Multimedia content attachment (size)	0	100KB	0	0	U
Supported attachment MIME types	-	image/jpeg, audio/wav, etc.	-	-	all
Notification lifetime control	0	0	1	0	0

* = larger messages are split into several messages; U(d) = Unlimited (device dependant)

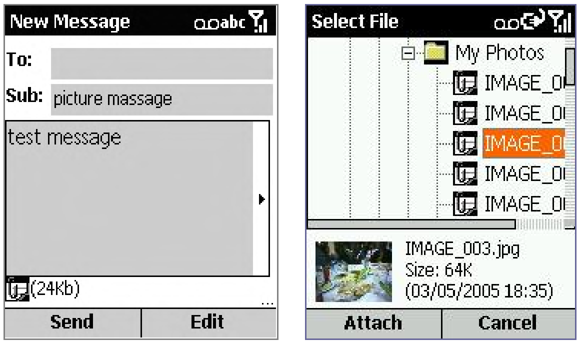
Table 2. Communication Channel Attributes (Selection)

Communication Channel *CC*	Cost function
SMS (GSM)	$C_{SMS/GSM} = ceiling\left(\frac{message\ size}{160\,bytes}\right) price_{SMS}$
MMS (GPRS)	$C_{MMS/GPRS} = \begin{cases} price_{MMS,low} & if\ message\,bytes + attachment\,bytes < 30KB \\ price_{MMS,high} & if\ 30KB \leq message\,bytes + attachment\,bytes \leq 100KB \end{cases}$
WAP Push SL (GSM)	$C_{WAPPushSI/GSM} = ceiling\left(\frac{message\ size}{160\,bytes}\right) price_{SMS}$
Email (Ethernet)	$C_{Email/Ethernet} = price_{Min}$ with $price_{Min} > 0$

Table 3. Communication Channel Cost Functions (Selection)

MESSAGING CLIENTS AND FRONT-END SERVICES

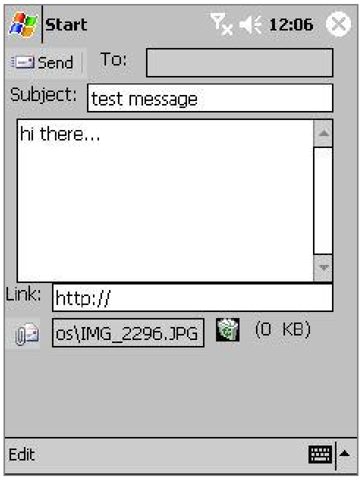
In order to overcome the constraints of the standard, channel specific messaging applications, we designed new messenger applications for different devices such as smartphones (Figure 3), PDA's (Figure 4), and desktop computers (Figure 5).



To:
Sub: picture massage
test message
(24Kb)

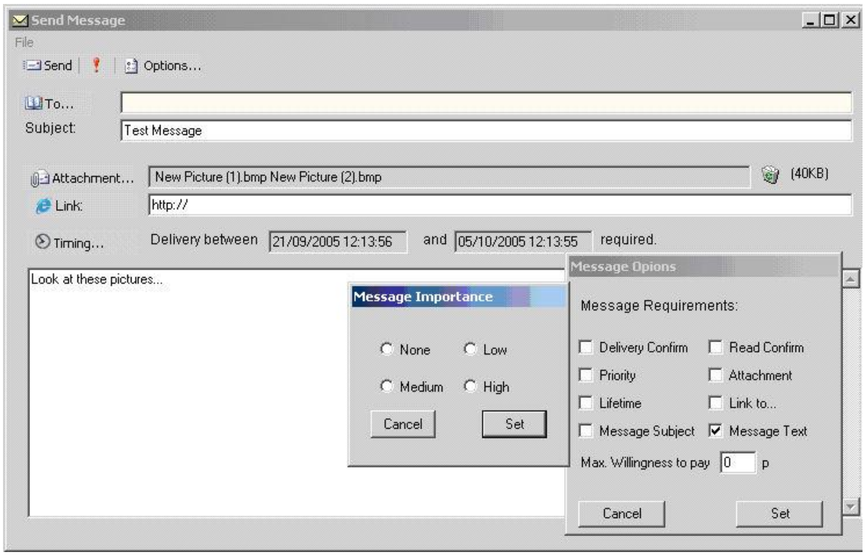
IMAGE_003.jpg
Size: 64K
(03/05/2005 18:35)

Figure 3. mCommunicator Smartphone Client (Windows Mobile 2003)



To:
Subject: test message
hi there...
Link: http://
os\IMG_2296.JPG (0 KB)

Figure 4. mCommunicator PDA Client (Windows CE)



To...
Subject: Test Message
Attachment... New Picture (1).bmp New Picture (2).bmp (40KB)
Link: http://
Timing... Delivery between 21/09/2005 12:13:56 and 05/10/2005 12:13:55 required.

Look at these pictures...

○ None ○ Low
○ Medium ○ High

Message Requirements:
☐ Delivery Confirm ☐ Read Confirm
☐ Priority ☐ Attachment
☐ Lifetime ☐ Link to...
☐ Message Subject ☑ Message Text
Max. Willingness to pay 0 p

Figure 5. mCommunicator Desktop Client (Windows XP)

These prototype applications allow the user to compose the content of a ‘generic’ message and receive recommendations how to send it. They run on different platforms, including MS Windows (desktop computers), Windows CE (PDAs) and Windows Mobile (smartphones). They all make use of the .NET compact framework and provide user login , message composition, specification of message attributes, and sending of the message to the server. In addition to standard features, the Smartphone enables the user to record a voice message and send it as a sound file attachment.

Client/Server Communication with Web Services

Communication between the mCommunicator server and the mCommunicator clients is realized via Web services since they provide access from both mobile and desktop clients using standard Internet protocols (Mallick, 2003). The mCommunicator server comprises two web services, one for user management and the other for message processing. These services include several functions, as depicted by the UML activity diagram in Figure 6. The following are seven steps that the system performs:

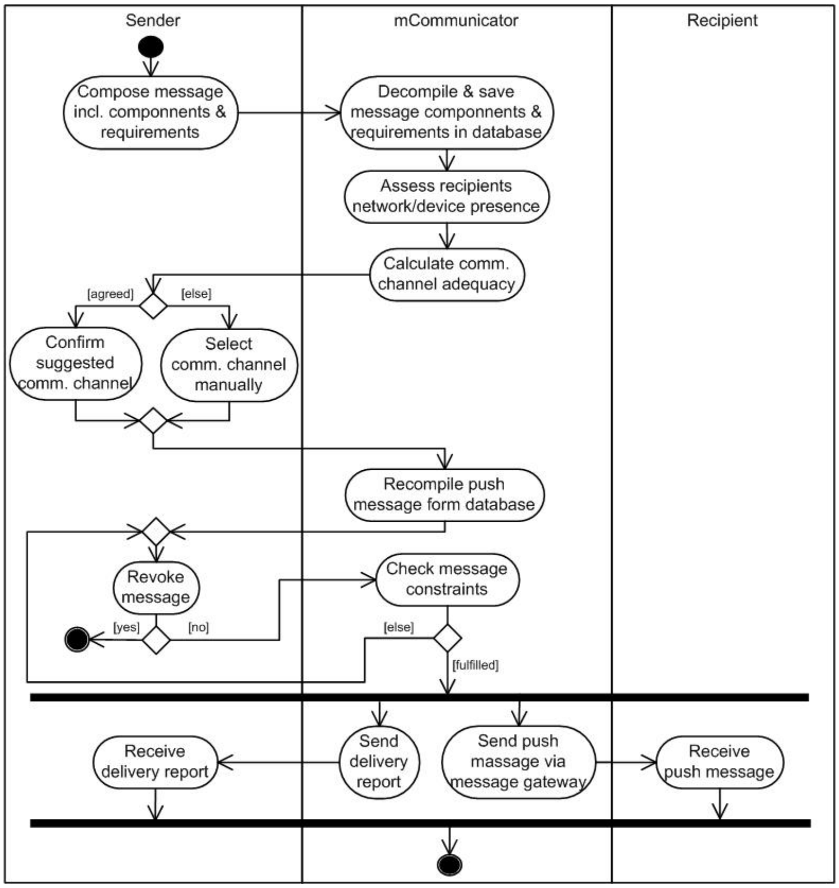


Figure 6. mCommunicator Interaction UML Diagram

- *Step 1.* First the sender compiles content of the message. That may involve text, voice recording, and file attachments. The messaging clients enable the user to define message delivery requirements such as general message attributes and maximum delivery costs. The user can also specify the time of delivery and can revoke the message any time before the actual delivery to the recipient.
- *Step 2.* After sending the message content to the mCommunicator server, all message data is stored in a database. Depending on the chosen communication channel, the appropriate message is recompiled from this data.
- *Step 3.* The mCommunicator server accesses information about the recipient and the recipient’s registered communication channels. It uses this information to provide the ranking of possible communication channels.
- *Step 4.* Based on the recipient status, the channel information, the sender’s preferences, and the message characteristics, the service ranks the available channels. It presents the user with the list of channels, indicating the corresponding utility

value and the message processing costs.

- *Step 5.* The sender selects one or multiple of the ranked channels. The sender can confirm the use of the communication channel proposed by the service or select one or more alternative ones from the list.
- *Step 6.* After the communication channel has been chosen, the service recompiles the content of the message into appropriate format. For example, if the user chooses to send an SMS the content will be stripped of attachments and message title; only message text will be delivered.
- *Step 7.* The mCommunicator server sends the message via the appropriate message gateway. If requested, the server notifies the sender about the status of the message. In case the user specifies the delivery time, the server provides the message delivery status as appropriate.

ARTIFACT EVALUATION

Since evaluation of designed artifacts is an iterative and incremental process that includes feedback loops (Markus et al., 2002; Hevner et al., 2004), we use our artifact instantiation as a source of valuable insight. We compare its characteristics with the functional requirements in Table 1 and summarize our evaluation criteria in Table 4.

Evaluation Criteria		**Corresponding design component/characteristic**	**Criteria satisfied**	**Design problem compliance (further system or evaluation requirements derived)**
System specific criteria (derived from problem statements)				
(1)	Integrated message editing	Message client design for editing channel-agnostic messages	yes	Messaging clients provide functionality for generic message editing.
(2)	Message roaming over IP and mobile networks	Server infrastructure design including different messaging gateways such as GSM and Email	yes	System infrastructure design provides possible connection to different message gateways (currently GSM & Email supported)
(3)	Message type conversion	Supported message components are recompiled from the origin message components	yes	The message compilation features the maximum set of components supported by the selected communication channel
(4)	Selection of most appropriate messaging service for sending a message	Comm. Space Model for recommending channel order based on message characteristics	partly	The model covers the senders' utility only. A more comprehensive model, assessing the total utility is desirable. More information regarding recipients' preferences and network capacity is required.
(5)	Information regarding recipient connectivity	Subscription of user devices to the mCommunicator service to record user device connectivity	partly	Manual service subscription is required.
General criteria				
(6)	System security requirements	User management for providing user accountability	partly	Used basic SQL security for client management. Further system analysi regarding security requirements is necessary
(7)	Ease of use	HCI design of messaging clients and client server interaction	partly	System provides integrated message editing. Further evaluation with user field or case study is required.

Table 4. IT Artifact Evaluation Results

We derive evaluation criteria from the functional requirements in Table 1. For example, the criteria *Integrated message editing* (Table 4) has been derived from the functional requirement *(1) Universal message client* (Table 1) which, in turn, stems from the use of proprietary formats and lack of interoperability among messaging clients. We perform architecture analysis in order to understand to which extent our artifact complies with these criteria. Furthermore, we identified two general IS evaluation criteria ((6) and (7) in Table 4) that also need to be addressed.

Based on the results of the evaluation process (Table 4), we note that some of the raised problems are well addressed with our approach. However, several issues are not solved completely ((4)-(7), Table 4) and raise questions for further research. Through the process of design, implementation, and analysis we gained experience that will help us modifying the artifacts and conduct more sophisticated evaluation. In particular, we plan user studies to assess the utility of the used recommendation model, the ease of use, and the social implications of the flexible, cross-service communication. Two particularly important aspects are user privacy and system security.

SUMMARY AND FURTHER RESEARCH

In this paper we identified issues and opportunities for innovation in helping users benefit from multiple communication channels, across a variety of platforms, devices, and distinct communication networks. We follow the guidelines of the design science research to explore the technical feasibility of providing an innovative solution. We apply iterative design and implementation and aim at evaluation in terms of "…functionality, completeness, consistency, accuracy, performance, reliability, usability, fit with the organization, and other relevant attributes" (Hevner, March, Park 2004).

We focused on achieving a channel-agnostic messaging, facilitated by the mCommunicator recommender system. The system also converts messages into the adequate message format and provides feedback and status information to the sender. We analyzed the resulting artifacts in relation to the functional requirements and discuss the strengths and drawbacks of our approach. Our effort provides evidence that issues are complex and further research and development are required to address the remaining problems.

Our next step will involve the second round of design and implementation in preparation for user studies. We plan to explore human computer interaction (HCI) issues and test the effectiveness of the recommendation model. We expect the user studies to reveal the perceived utility and appropriateness of different communication channels and the channel recommendation model itself (Westmyer, DiCioccio and Rubin, 1998). From these additional studies we hope to learn more about the messaging communication processes and arrive at clear guidelines for building more effective artifacts and production systems.

REFERENCES

1. Au, Y. A. (2001) Design Science I: The Role of Design Science in Electronic Commerce Research, *Communications of AIS*, 7, 1, Article 1.
2. Hevner, A. R., March, S. T. and Park, J. (2004) Design Science in Information Systems Research, *MIS Quarterly*, 28, 1, 75-105.
3. King, R. and Xia, W. (1997) Media appropriateness: Effects of experience on communication media choice. *Decision Sciences*, 28, 4, 877-910.
4. Mallick, M. (2003) Mobile and Wireless Design Essentials, Wiley, New York.
5. Maniatis, P., Roussopoulos, M., Swierk, E., Lai, K., Appenzeller, G., Zhao, X. and Baker, M. (1999) The Mobile People Architecture, *ACM SIGMOBILE Mobile Computing and Communications Review*, 3, 3, 36-42.
6. March, S. T. and Smith, G. F. (1995) Design and Natural Science Research on Information Technology, *Decision Support Systems*, 15, 4, 251-266.
7. Markus, M. L., Majchrzak, A. and Gasser, L. (2002) A Design Theory for Systems that Support Emergent Knowledge Processes, *MIS Quart*erly, 3, 26, 179-212.
8. Moon, J. and Sanders, G. (2004) Lawrence. The Usefulness of Interpersonal Communication Media: the synchrony and symmetry model and the media richness theory revisited, in Christine Bullen and Edward Stohr (Eds.) *Proceedings of the Tenth American Conference on Information Systems*, August 6-8, New York, NY, USA, 1578-1584.
9. Nakanishi, Y., Kumazawa, S., Tsuji, T. and Hakozaki, K. (2003) iCAMS2: developing a Mobile Communication Tool using Location Information and Schedule Information with J2ME, in Luca Chittaro (Ed.) *Proceedings of Mobile HCI 2003*, September 8-11, Udine, Italy, 400-404.
10. Nunamaker, J. F., Chen, M. and Purdin, T. D. (1991) Systems Development in Information Systems Research, *Journal of Management Information Systems*, 7, 3, 89-106.

11. Orlikowski, W. J. and Lacono, C. S. (2001) Research Commentary: Desperately Seeking the 'IT' in IT Research - A Call to Theorizing the IT Artifact, *Information Systems Rese*arch, 12, 2, 121-134.
12. Simon, H. A. (1996) The Sciences of the Artificial, The MIT Press, Cambridge.
13. Srinivasan, A., March, S. and Saunders, C. (2005) Information Technology and Organizational Contexts: Orienting Our Work Along Key Dimensions, in David Avison, Dennis Galletta and Janice I. DeGross (Eds.) *Proceedings of the Twenty-Sixth International Conference on Information Systems*, December 11-14, Las Vegas, NV, USA 991-1001.
14. Tang, J. C., Yankelovich, N., Begole, J., Van Kleek, M., Li, F. and Bhalodia, J. (2001) ConNexus to Awarenex: Extending awareness to mobile users, in Julie Jacko and Andrew Sears (Eds.) *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, March 31-April 5, Seattle, WA, USA, 221-228.
15. Tiwana, A. and Ramesh, B. (2001) A Design Knowledge Management System to Support Collaborative Information Product Evolution, *Decision Support Systems,* 31, 2, 241-262.
16. Westmyer, S. A., DiCioccio, R. L. and Rubin, R. B. (1998) Appropriateness and Effectiveness of Communication Channels in Competent Interpersonal Communication, *Journal of Communication*, 48, 3, 27-48.