Designing User Interfaces for Mobile Business Processes using Messaging Application

Lasse Pajunen, Suresh Chande, Sameh Galal

Nokia Research Center
Nokia Corporation
P.O.Box 407, FI-00045 Nokia Group, Finland
lasse.pajunen@nokia.com
suresh.chande@nokia.com

Stanford University
621 Escondido Road, Stanford, CA 94305
sameh.galal@stanford.edu

Abstract: Mobile devices have user interfaces which enable users to interact with each other in an intuitive manner. The most popular applications are voice calls and messaging. In contrast, business communication requires support for long-running interactions between people and automated systems. This makes the use of mobile device for business communication complicated with the existing user interface capabilities. Especially, in cases of mobile device centric processes, intuitive and effective mobile user interface is needed. A mobile business process is sub class of general business processes where user interaction is performed using mobile devices. In this paper, we will describe our experiences in enhancing the existing messaging platform and providing an intuitive way for users to access and interact with business processes. We will describe our approach and a proof-of-concept system with convenient and natural user interface that compares to the other common functionalities of the mobile device. The system consists of four parts: seamlessly integrated way to start new mobile business process instances, ways to check status of existing process instance execution, ways for notifying user on external events within a process, and ways on how users can react on these external events.

1 Introduction

In business processes, people and systems cooperate to fulfill some business goals. Mobile business process is a sub class of general business processes where most of the human interaction is done using mobile devices such as smart phones. Typically such a business process integrates fully automated activities from computer systems and non-automated activities executed by people. These human executed activities in business processes are also called as human tasks.

Mobiles devices are nowadays used to support and perform some of human tasks. Especially, people are used to receive emails and use web browser based systems in order to perform some mobile interaction with business processes. However, web browser based interfaces are often not intuitive and useful because mobile devices are mainly used in asynchronous mode. Compared with the online web access model, messaging access is more preferred in the wireless environment because messaging systems work in an asynchronous way and can tolerate intermittent connectivity [Ju05].

Nowadays, more and more mobile devices are used to interact with automated business processes. Previously business processes have been developed with personal computer based interfaces due to their availability. However, the recent explosion of mobile computing devices such as smart mobile phones has changed the situation. Additionally, use of mobile devices have enabled new opportunities as well as. Therefore, business infrastructure has started to support mobile workforce who typically work outside of traditional office environments and need to access the organizational processes in order for being productive. To begin with, the mobile devices are used to interact with traditional processes. However, it will require new models on how the user interaction is designed and implemented in order to leverage fully the benefits of mobility.

Business processes are automated using workflow engines with description languages such as Web Services Description Language (WSDL) and Web Services Business Process Execution Language (WS-BPEL). Web service standards (WS-*) and Service-Oriented-Architecture (SOA) enable a common framework for system integration. This framework provides protocol level integration between systems allowing all components to exchange information with each other. Other similar frameworks for business to business process automation have been proposed by standards developed by WfMC, ebXML and RosettaNet.

However, human tasks and human interaction are not covered by aforementioned integration technologies, and there are few solutions on how these technologies are linked with user interfaces. Often, user interfaces are either provided by specialized applications or by general web browser based solutions. Web browser based solutions enable a standards based approach on user interface design, but this approach is still lacking a standardized mapping between the automated process and user interface description. Typically, this mapping is implemented using different scripting languages such as PHP: Hypertext Preprocessor (PHP) and Practical Extraction and Report Language (Perl).

In this paper, we will analyze use of business processes (especially mobile business processes) and identify main scenarios for user interaction. Then we will propose how this previously described gap between business process and user interaction can be fulfilled. In addition, we will describe mobile process in a relationship with other mobile processes and networked processes.

The rest of the paper is organized as follows. In Section 2, we will analyze other related ways to integrate human tasks in the business processes. In Section 3, we will further analyze user interface needs in business processes. In Section 4, we will describe how messaging application can be used to provide a convenient way to integrate to business processes. In Section 5, we will provide details how user interface and business process can be integrated. In Section 6, we will describe our proof-of-concept implementation. And finally in Section 7, we will conclude our paper and provide ideas for future work.

2 Related Work

In this paper, we will analyze user interface needs in mobilizing business process. We start by reviewing related research. Much research has been done on how to visualize information on mobile devices with restrictions on screen size etc. However, less research is done on how to start participating with the processes and how the processes can inform users on some events.

In our recent research on ActiveForms [CP07], we have developed a runtime environment which enables user to interact with process oriented workflow engine by means of standard client environment by isolating the user interface with the process logic. We have seen similar research such as iMMS framework where a similar view towards "compared with the online web access model, messaging access is more preferred in the wireless environment because messaging systems work in an asynchronous way and can tolerate intermittent connectivity" [Ju05].

As noticed in the iMMS framework [Ju05], the goal has been to enhance the mobile user interface and develop an underlying system to enable a conversational communication between mobile users involved in business processes. The research carried out by iMMS framework depends on extension to standards such as web browsers and depends on existence of specific backend systems compared to our research. We have realized the approach where iMMS framework uses the embedded media in MMS client is confusing and cumbersome to use. This was found out in user studies and research carried out in forms based structured messaging framework [Ch06]. As a result of the study, ActiveForms [CP07] was redesigned with an integrated user interaction method to provide a better and more natural interface to the user.

While developing support for different devices in various environments, we have also learned that mobile processes are often parts of more complicated composed processes. Therefore, sometimes, it is useful to separate user interface dependent and independent parts in business processes. [YP07]

Visualizing information on mobile devices is a well researched topic. There are many papers covering adaptation issues with small screen and limited network bandwidth. For example, if the bandwidth is limited, the data sent to the user must be focused [CSD01, MLW98]. Focusing here means that only really required information is sent, but also with a guarantee that all needed information is really sent. This is also very relevant to the role and task based interfaces in the process. We are focusing by sending only relevant data in the messages, but further work could include support for image resolution adjustments. In the same way, the user interface must be designed based on the screen size and other device capabilities [KR03, Ni04]. When the screen size is limited and available input controls differ, extra care must be put in taking all capabilities to use in a maximal way and providing the user a possibility to do the most common things in an easy way. Here, we support different user interface templates for each platform.

Another way to think of all user interfaces of the whole process is to consider them as one distributed interface of the whole application. In here, pieces of user interfaces can be modeled and executable pieces composed [DCS05]. The limitation in this approach is a lack of knowledge of dynamic issues in process execution. Therefore, the approach is better suited for static interaction.

In a similar way, providing user interfaces for process has been a topic of many research projects. PerCollab [CH04] is a system that integrates business processes and various user interface technologies. In there, an Interaction Controller manages all interactions with the user. This approach puts much weight on the Interaction Controller. Therefore, this approach is suitable when the user interaction is simple, a basic request (for example filling one form) from the user. However, we see that more complex user interactions are often needed.

Other similar mobilizing studies have been made. GreenBSN [LW05] and WOSE workflow framework [HWR05] are examples of systems, where a gateway is used to provide a most suited mobile interface for the device and to find a most suited service for the situation. The fourth similar concept is Lynx [VV05]. Here, the user interaction is implemented on top of an email system. All these kinds of approaches work well for similar cases such as PerCollab.

3 User Interface Needs in Mobile Business Processes

In this section, we will analyze user interface needs in mobile business processes. We will first analyze a reference scenario based on task assignment use case. Then we will analyze technologies needed to fulfill user interface needs in the scenario. And finally, we will analyze needs to processes to provide previous interfaces.

First, we will analyze the difference between mobile sub user interface processes vs. the whole process with multiple cooperating sub-processes. The basic assumption is that the whole business process composes many people and many automated systems. Therefore, there can be multiple WS-BPEL-based processes running in parallel. Mobile sub user interface process is a WS-BPEL-based process that is executed in one mobile device. When all of the sub processes and other services are collaboration, the whole business process is proceeding.

In Figure 1, a business process consists on three cooperating processes. Manager's and worker's processes are executed in mobile devices and organization's process is executed in the server. Mobile processes focus on user interaction. Server side process focuses on integration to other company systems. For example, in task assignment and validation assignment, the human resources services are used. In sending invoice the accounting services are used. To introduce mobile devices as a main business tool that participate in business processes, these devices have to have their workflow processing engines that can understand the different semantics of business logic using languages such as WS-BPEL.

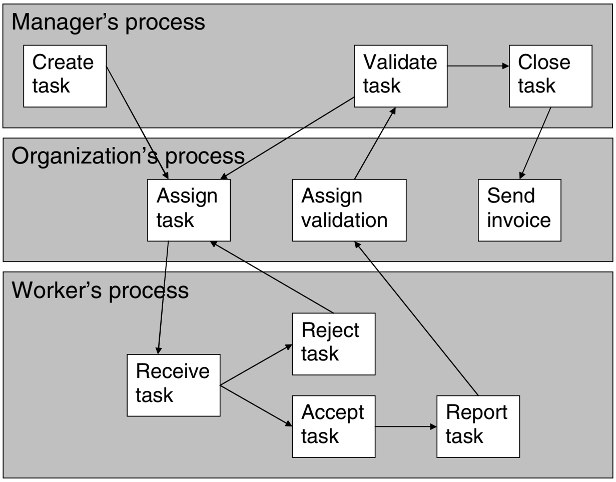


Figure 1: Task assignment business processes consisting of three cooperating processes

3.1 Reference Scenario

Figure 1 describes a business process in abstract terms. In practice, processes include more details and activities. This means that individual activities (represented as boxes) consist of sub activities.

As a use case, we will analyze a business process for a remote task execution such as painting of a house. In this use case, a worker accepts a task request. However, before actually performing it, the worker must first travel right location with support of a map. After finishing the work, a form must be filled. In the form, information such as current weather condition, current distance meter value from the car, how much resources where needed to fulfill the task, and how much time was spent are asked. After the worker fills the form, the task execution finally ends.

Typically, interacting with a business process is not a continuous slot of time but a set of slices. For example, after receiving a task, the worker needs to travel, maybe by driving with a car. While driving, there may be breaks in the process, when the worker has a coffee break or lunch.

In mobile and email messaging, the breaks are often supported by allowing saving unfinished messages to drafts. In a same way, in business processes, we need a way to break participation and then allow continuing the work.

3.2 User Interface Components

In previous reference scenario, we see that different user interface components are needed. In addition to the actually performing the task (for example, by filling a report), we need ways to start and stop the work. For this, there are four cases. First, we need a way to start new process. Second, we need a way to break and continue the process. Third, we need a way to notify on external event, and finally, we need a way to close the process.

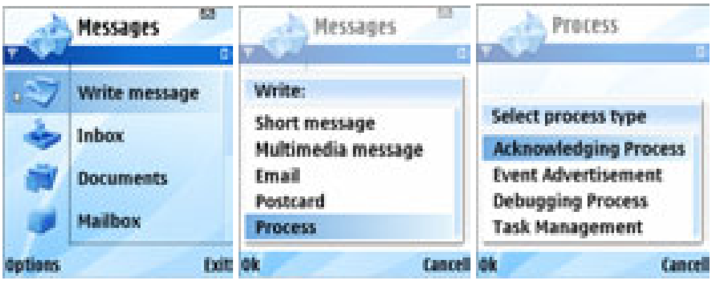
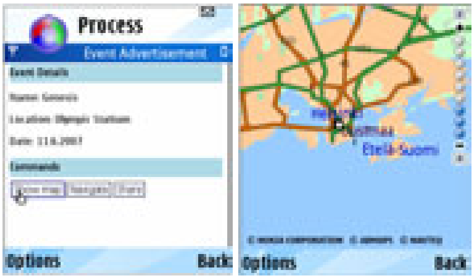


Figure 2: Initiation of a new process from messaging application

Figure 2 shows user interface on how process can be initiated. The basic requirement for process initialization component is a selection of initiated process. Typically, there are many deployed processes and the user must be able to select which one is needed.



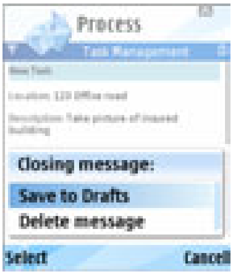
Name: Genesis

Location: Olympic Stadium

Date: 11.6.2007

Figure 3: Performing some tasks

After initialization, some tasks needs to be performed. Figure 3 visualizes two different used interfaces that can be used in actual task execution. In the left side of the figure, a form based interface can be used to visualize process data and allow making decisions. On the right side, a map and navigation based user interface is shown. Therefore, a user is allowed to get location information related to the task.



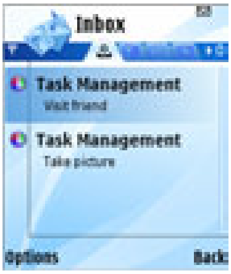
New Task

Location: 123 Office road

Description: Take picture of injured building

Figure 4: Saving to drafts

Sometimes the whole task cannot be performed immediately. In these cases, there must be a user interface component to store an unfinished task. In Figure 5, a sample user interface shows how the message or unfinished task can be saved to drafts. In some cases, the user may notice that this task can never be finished. In order to support closing the task completely, in the figure, the user can select if the task is saved or completely discarded.



Task Management
Visit friend

Task Management
Take picture

Figure 5: Notifying the user on external event

The user interface must contain a component for finding previously saved unfinished tasks, happened external events etc. Figure 4 shows a way to notify the user on external event. Often, there are lots of unfinished tasks, external notifications etc. Therefore, there should be user interface for searching or categorizing different types of items. In the figure, this kind of categorization is used. Inbox category contains notifications on external events. In addition there are separate categories for drafts and sent (executed) messages.

Finally, there needs to be a user interface component for closing and deleting process related notifications. This should be done in the same interface as the other finding and selecting process messages. For this purpose in user interface there should be an option for closing from the application or deleting messages from the list.

3.3 User Interface Process

The end user process participation can be divided in seven phases. In this section, we will describe these phases in more detail. However, the basic concept here is that users are in this phases and mobile business processes must support the user in each of the phases in a phase related way. The phases are:

1. Starting a new process instance.
2. Initial interacting with the process.
3. Breaking for future continuation.
4. Waiting for some external event to occur, waiting some time to occur or reaching to a particular location.
5. Continuing process.
6. Checking status of process instances.
7. Closing the process instance.

The first phase is initiation. In this phase, the user should be able to query available processes and then initiate a selected process. This selection can be made from travel expense claim process, report writing process etc.

After initiation, the second phase is initial interacting with the process. Often processes require some basic configuration or setting up. For example, travel claim process requires filling a travel expense form. In this form, basic questions such as travel destination, length, expenses are asked. After the initial interaction, the process can really start, and other participants to the process can be engaged.

After initial interaction or after continuing the process later, there might be a need to break for future continuation. This is often the case, if the process requires some external activities to happen. After break the process interaction, the user goes to waiting phase.

In waiting phase, a user is waiting for some external event to occur, some time to pass, or to reach to a particular location. The basic requirement here is to be able to notify user after continuation is possible. For example, this can happen after a manager has approved a request.

After the user gets a notification, the process interaction should be continued. The continuation depends on the process. In some cases, the process requires input or choices to be made by the user. For example, in a process that requires the user to approve some request that arrives to the mobile device. In some cases, the continuation is only a seeing the outcome of external activities, such as seeing the result of approval request.

Optionally after waiting phase, a checking phase can occur. In this case, user has not received a notification, but there is anyway a need to check the status and maybe do some actions. This checking phase is in a way an interaction phase between waiting phase and continuation phase.

Finally, there is a process closing phase. Basically, this can happen after any phase. Due various reasons, the user may want to close the process after some interaction or while waiting.

4 Messaging Application based User Interfaces

Users have grown used to the messaging centric user interfaces with their mobile devices since main uses of mobile device have been voice calls and message exchanges. Current mobile devices support several messaging technologies such as Short Message Service (SMS), Multimedia Messaging Service (MMS) and email. In addition, the messaging framework is often extensible for new custom message types. The user interface for the messaging application has undergone a long evolution and its usability has been improved remarkably over the years. Moreover, users and especially business users have grown accustomed to using the messaging application especially for emails extensively.

The main functions of Messaging application are:

- Start writing new message
- Store, see, and continue message in Draft box
- Store and see message in Sent box
- Receive and see message in Inbox
- Update information in any of these boxes
- Removing messages from any of these boxes

In addition, the messaging application is well integrated with other mobile device user interfaces. The messaging application is quite good option for user notification of events, as it can display small indicator in almost every screen on the mobile device if there is a new unread messages in the inbox. The messaging application also offers the user to create drafts of their messages to be edited and sent later. And finally, the sent folder is used to find information on events that has happened before.

Therefore, messaging application has a very good match when comparing requirements for mobile process user interfaces. The messaging application has a way to create, input, break, open, continue, update, and send messages. Since the mobile business processes paradigm is very close to message exchange, using this message exchange paradigm and messaging application fits well for mobile business processes.

A mapping between messaging concepts and process concepts is straightforward. First, there is a link between a message in messaging and process instance in processes. Then secondly, there is a link between message type and process type.

Then we can have a mapping between concept attributes. In a message there is a sender. In process notifications, there is a sending process. In messages, there is a subject. In process notifications, there is a process instance description. And finally, in messages and process notifications, there is an attribute if they are read or not.

The final mapping is in categorization. While incoming notifications map to incoming messages, they will be stored in inbox. Executed actions allowing status queries map to sent messages, and therefore they will be stored in sent folder. Unfinished tasks map to unfinished messages and they will be stored to drafts folder.

As a summary, we can see that all process phase needs can be mapped directly to message exchange concepts in messaging application.

5 Integrating Mobile Business Process and Messaging Application

As introduced earlier, the messaging user interface of mobile devices could be a very good candidate for fulfilling the different user interface requirements of the workflow processing engine as such:

- Process initiation: The user will be able to see a new process option as one messaging options in addition to SMS, MMS and email when choosing to write a new message

- User notification: the workflow engine can create new messages in the inbox and use the existing messaging facilities such as the beeping sound when a new message arrives to alert the user

- User interaction: Additionally after the user opens a message that the workflow engine has created, he will have the options to interact with the process and input his data and decisions

- Status display: Within the messaging application, there can be an entry set up to look at the status of current processes

The last remaining question is how to use messaging application for the WS-BPEL based mobile process. The answer is to model messaging application as a web service and allow mobile workflow engine to send and receive messages from it.

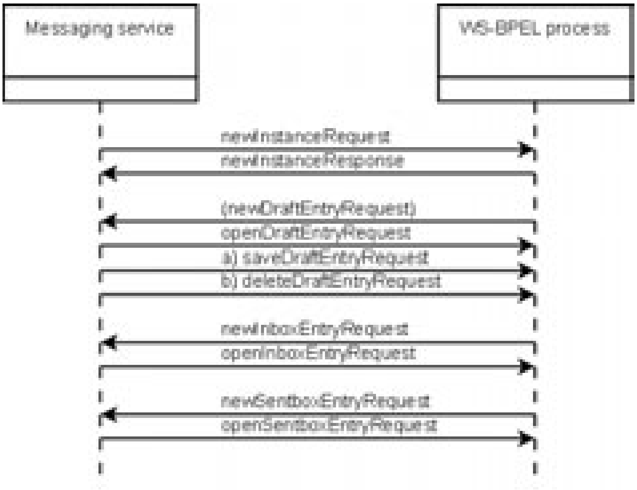


Figure 6: Communication between Messaging service and WS-BPEL based process

Figure 6 shows message exchange between Messaging service and WS-BPEL based process. It is to be noted that all operation groups can happen many times. newDraftEntryRequest may be replaced by newInstanceResponse. Cases a, b or neither of them can happen after openDraftEntryRequest.

6 Implementation Experiences

We have developed a proof-of-concept prototype to test and to demonstrate use of mobile business processes. This prototype consists of a set of sample business processes and a mobile workflow engine that is integrated to various mobile device applications, especially to messaging. We have analyzed sample processes that run on a single device and processes where multiple devices cooperate to achieve a shared goal. In all of these cases, use of messaging as a basis for user interfaces has been found useful.

Integration of the mobile workflow engine to the messaging application in the mobile device is facilitated by the Nokia S60 platform which is based on the Symbian mobile operating system. The integrating is based on Symbian messaging architecture, which is a multi-protocol messaging framework. This framework supports creation and installation of plug-in modules to support individual messaging protocols such as fax, e-mail, SMS, and MMS. Each plug-in module is called a Message Type Module (MTM). The framework allows messaging to be used as an integrated user experience rather than as a series of separate applications. The messaging provides a unified user interface for the different message types, which allows creation, editing, sending, replying and forwarding of new messages depending on protocol type [Tu02].

We have implemented the needed user interface module for the workflow engine as an additional plug-in as a MTM component to the messaging application. This allows the workflow engine to benefit the powerful user interface features of the messaging application and allows it to be integrated seamlessly to the device user interface. For example, when the user opens an image or takes an image using camera in mobile device and selects to send this image, the user has the option to send this image directly using the mobile workflow engine. As such, ease of use and visibility of the process capabilities are very strong assets of this approach to user interface integration of mobile business processes.

The implemented MTM module enables support for all processes deployed to the mobile workflow engine. Therefore, each process does not need an own MTM module. This support for multiple processes using one MTM module is enabled by allowing the user to choose the process type when creating a new process instance. The options for this selection are dynamically queried from deployed processes. For example, the user might be asked to choose between travel expense claim process, meeting process, and inventory request process.

As described in Section 5, the mobile workflow engine-messaging integration is bidirectional. In some situations, the messaging plug-in sends a message to the mobile workflow engine and some situations, the mobile workflow engine sends a message to the messaging application. The messaging framework supports both situations and our prototype uses them both.

In addition to implementing our prototype, we have tested the prototype successfully within our research team. We have tried a couple of imaginary scenarios. In the future, we would like to trial this technology with real users in real business scenarios.

7 Conclusions and Future Work

Users view their mobile devices as their main communication devices. Because business processes are highly distributed applications requiring communication between participating parties, the use of messaging user interface for hiding complexity of business processes seems to be the perfect match for convenience, usability, and user perception. Users will tend to view their interaction with process mainly as a communication activity with other people participating in these processes. Furthermore, no coaching and training are needed for people to start using the system, since virtually everybody who knows how to send and receive SMS messages. Therefore they should be able to use the process messaging immediately because it uses the same paradigm.

We have developed an integrated business process system in the manner that it can provide opportunities to rethink tradition process oriented systems. However, in order to better understand the impact to end users, we see need for further studies in real deployments and new work environments where personal computers do not exists and where in most cases people are using mobile as their only computers. We expect to carry this kind of testing in new mobile work environments, such as in rural India, Africa, where the mobile device are potentially the only computer used to run ones own businesses.

Human interaction is often missing part with business process and integration technologies. However in this paper, we have shown that mobile business processes can use messaging application after it has been turned into a service. This enables support for human interaction with standard technologies. By using web services, this can be done by describing functionality of messaging application using WSDL and providing a service implementation that allows external communication using SOAP messaging. Then messaging can be directly used in WS-BPEL-based processes.

References

[CH04] Chakraborty, D. and Hui L.: Pervasive Enablement of Business Processes. Pervasive Computing and Communications, 2004. PerCom 2004. Proceedings of the Second IEEE Annual Conference on 2004 Page(s):87 - 97 Digital Object Identifier 10.1109/PERCOM.2004.1276848

[CSD01] Chalmers, D., Sloman, M., and Dulay, N.: Map adaptation for users of mobile systems. In Proceedings of the 10th international Conference on World Wide Web (Hong Kong, Hong Kong, May 01 - 05, 2001). WWW '01. ACM Press, New York, NY, 735-744. DOI= http://doi.acm.org/10.1145/371920.372193

[Ch06] Chande, S.: Form based Structured Mobile Messaging Framework for Mobilizing Business Applications, COMSWARE 2006, New Delhi, India.

[CP07] Chande, S. Pajunen L.: Active Forms: A Runtime for Mobile Application Forms, The Sixth International Conference on Mobile Business (ICMB) 2007, Toronto, Canada

[DCS05] Demeure, A., Calvary, G., Sottet, J., and Vanderdonkt, J.: A reference model for distributed user interfaces. In Proceedings of the 4th international Workshop on Task Models and Diagrams (Gdansk, Poland, September 26 - 27, 2005). TAMODIA '05, vol. 127. ACM Press, New York, NY, 79-86. DOI= http://doi.acm.org/10.1145/1122935.1122952

[Ju05] Jun, S. et al: Delivering mobile enterprise applications on iMMS framework, International Conference On Mobile Data Management, Proceedings of the 6th international conference on Mobile data management, 2005, Ayia Napa, Cyprus.

[KR03] Kaikkonen, A. and Roto, V.: Navigating in a mobile XHTML application. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Ft. Lauderdale, Florida, USA, April 05 - 10, 2003). CHI '03. ACM Press, New York, NY, 329-336. DOI= http://doi.acm.org/10.1145/642611.642669

[LW05] Liang, Z. L. and Wong, R. K.: A lightweight mobile platform for Business Services Networks. In Proceedings of the IEEE Eee05 international Workshop on Business Services Networks (Hong Kong, March 29 - 29, 2005). ACM International Conference Proceeding Series, vol. 87. IEEE Press, Piscataway, NJ, 12-12.

[HWR05] Huang L., Walker D. W., Rana O. F. and Huang Y.; Dynamic Invocation, Optimization and Interoperation of Service-oriented Workflow, Work-in-Progress Section, the 5th International Symposium on Cluster Computing and Grid Computing, Cardiff, Wales, U. K. May 2005

[MLW98] McIlhagga, M., Light, A., and Wakeman, I.: Towards a design methodology for adaptive applications. In Proceedings of the 4th Annual ACM/IEEE international Conference on Mobile Computing and Networking (Dallas, Texas, United States, October 25 - 30, 1998). W. P. Osborne and D. Moghe, Eds. MobiCom '98. ACM Press, New York, NY, 133-144. DOI= http://doi.acm.org/10.1145/288235.288270

[Ni04] Nikkanen, M.: User-centered development of a browser-agnostic mobile e-mail application. In Proceedings of the Third Nordic Conference on Human-Computer interaction (Tampere, Finland, October 23 - 27, 2004). NordiCHI '04, vol. 82. ACM Press, New York, NY, 53-56. DOI= http://doi.acm.org/10.1145/1028014.1028021

[Tu02] Turfus C.: Using MMS on Symbian OS phones - PART 1. Symbian Technical Library. http://www.symbian.com/developer/techlib/papers/mms/mms.pdf

[VV05] Velez, I.P. and Velez, B.: Lynx: An Open Email Extension for Workflow Systems Based on Web Services and its Application to Digital Government. Telecommunications, 2006. AICT-ICIW '06. International Conference on Internet and Web Applications and Services/Advanced International Conference on19-25 Feb. 2006 Page(s):160 - 160 Digital Object Identifier 10.1109/AICT-ICIW.2006.125

[YP07] Yang Y., Pajunen L.: Developing Agile User Interfaces for Heterogeneous Devices in Business Processes, ICEIS 2007, 9th International Conference on Enterprise Information Systems, June 2007.