Personalisation of Future Mobile Services

[1]Ivar Jørstad, ivar@ongx.org, [2]Do Van Thanh, thanh-van.do@telenor.com, [3]Schahram Dustdar,
dustdar@infosys.tuwien.ac.at

[1]Norwegian University of Science and Technology, Dept. of Telematics, O.S. Bragstads Plass 2E, N-7491 Trondheim, Norway,
[2]Telenor R&D, Snarøyveien 30, N-1331 Fornebu, Norway, [3]Vienna University of Technology, Distributed Systems Group (DSG),
Information Systems Institute, A-1040 Wien, Argentinierstrasse 8/184-1, Austria

Abstract
Personalisation has been regarded as an important feature of mobile services. Still, only few services are subject to personalisation, and the features that are personalised are limited. It is also not entirely clear what personalisation is and what it will mean for future mobile services. This paper discusses personalisation of mobile services in general. A definition is provided, and the various opportunities for personalisation are introduced. In particular, features and services that can or should be subject to personalisation are discussed, and also mechanisms to achieve this are described. The design of a solution realising these mechanisms using XML Web Services is proposed.

1.   Introduction

Personalisation of services on the WWW has already been extensively investigated from various viewpoints [1][2]. Also, personalisation has often been discussed in relation to mobile services, but a study of how personalisation differs for these services and potential technologies and solutions to address the concept is missing. In addition to this, it has been suggested that personalisation might be overrated [3]. A collection with comments and summaries of some literature on personalisation can be found in [4]. This paper starts by examining current services that are personalised, and identifies the categories of features that can be subject to personalisation, before describing challenges to realise personalisation features and proposing specific solutions to these challenges. Section 2 attempts to answer what personalisation is or could be by describing the motivation for personalisation, and the features subject to, personalisation. Section 3 discusses possible solutions to challenges with personalisation of mobile services. Finally, Section 4 gives a conclusion and suggestions for future work.

2.   What is Personalisation?

There exist several definitions of what personalisation is. In [5] it is defined as:
"Personalization and customization refers to the ability of an Internet Web site or service to be shaped or re-shaped so as to better meet the individual needs or wants of a user."

In our case, this would be adapted to concern mobile services instead of Internet Web sites or services. Sometimes, a distinction is made between customisation and personalisation [3], where customisation is thought of as user controlled modifications of a service and personalisation is machine-controlled (i.e. performed by intelligence in the network). However, the concept of personalisation means that something is individualised to fit a specific person's needs and does not include or exclude any specific mechanism to this end. Thus, in the following, customisation is merely regarded as *one* way to personalise a service. Throughout this paper, the following general definition of personalisation is assumed:
"Personalisation of a service means that mechanisms exist to allow a user U *to adapt, or produce, a* service A *to fit* user U's *particular needs, and that after such personalisation, all subsequent service rendering by* service A *towards* user U *is changed accordingly."*

The implications of this definition will be discussed throughout Section III of this paper, but initially it can be noted that the definition implies more user control of service behaviour than earlier definitions.

2.1   Motivation for Personalisation

In a taxonomy and psychology study of personalisation [6], the motivation for personalisation is divided into two major categories:
1. Personalisation to facilitate work
2. Personalisation to accommodate social requirements

True, measurable benefits are easier to see of category 1, as such personalisation can increase the efficiency of performed work. One example is to automate or simplify regular tasks and another is to adapt a service to people with physical differences (e.g. weak-sightedness etc.).
However, services that are personalised today are rarely of category 1, but rather of category 2. Examples are the unlimited number of ring tones and logos that are available for download to mobile handsets. Also, personalisation of ring-back tones has been a hot topic.
These two categories of motivation can be further subdivided into other categories. Personalisation to facilitate work can mean both to increase efficiency in everyday tasks for a busy businessman, but it can also

mean to *allow* an individual with physical differences to participate in particular tasks.

It is reasonnable to believe, as the opportunities to provide complex mobile services increases, personalisation will prevail as a highly requested feature.

2.2 Personalisation Features

With some knowledge of the motivation for performing personalisation of mobile services, what is lacking is a better view of what features can be subject to personalisation. For services on stationary computers (e.g. in Windows XP), personalisation often includes:

1. Changing background image
2. Changing theme (e.g. global title bar colour etc.)
3. Installing specific services
4. Adding personal information to installed services (e.g. MSN Messenger)
5. Changing organisation of installed services (order of desktop icons, access through Quick Launch bar etc.)

The question is then, which of these are applicable to mobile services, and what other types of features should be possible to personalise in a mobile context that might not be as important in a stationary context. We propose to initially categorise personalisation features into the following categories:

1. Look and Feel
2. Personalisation of Service Portfolio
3. Individual Service Personalisation (behaviour)

In addition to these, we introduce two additional categories of personalisation. The first one is *Personalisation by Content*. This is so elementary that by some it might not be regarded as personalisation. But in fact, for mobile services, it can be the most important personalisation feature of them all. Personalisation by content implies that a service has been personalised as soon as the use of it has resulted in the production of some content. For example, writing this paper results in a Word document, i.e., Word (as a whole, including the document) became a personalised service to the author as soon as the title of this paper was written. Considering the content as part of the service makes sense, because most services are useless without any content (documents).

The second additional category is *Personalised Service Composition*. Composition can be seen to consist of those activities required to combine and link existing services (atomic and composite services) and other components to create new processes [7]. A formal treatment of service composition can be found in [8], where composition of agents (processes/service components) is accomplished by combining compatible inputs and outputs. For the particular case of Web Services, *service composition* is the process of combining the functionality provided by Web Services *components* to form a *composite* Web Service.

Dynamic service composition has previously been subject to some research [9], but the personalised service composition is a concept where *end-users* can perform such composition at runtime (see Figure 1). This results in new requirements to the service architecture, as will be seen later in this paper.
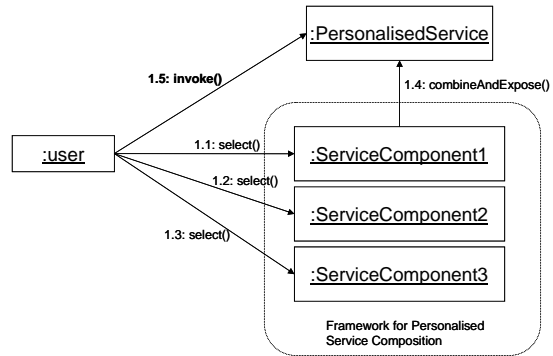

Figure 1: Personalised Service Composition.

One example is to combine a user profile service with a directory lookup service (e.g. a bank account listing), where the lookup key is based on the information taken from the user profile. The resulting service, which can be exposed as a complete service by itself, is personalised and will have meaning only to this particular user. By performing such service composition, access to services can be more effective and less cumbersome (especially if the information taken from e.g. the user profile is extensive).

Of course, all services that records user behaviour and user information in a profile will be concerned with privacy issues. One study of personalisation in m-commerce [10] suggests that privacy is important, but may not be important enough if the benefits of personalisation are sufficiently high and easy to see for the user.

2.3 The General Process of Personalising a Service

There are in general two different mechanisms for applying personalisation to a service; *explicit* or *implicit*. If personalisation is achieved explicitly, it means that one part (either user or provider of service) "tweaks" the parameters of the service manually. In contrast, with implicit personalisation, mechanisms that are connected with the service more or less continuously adapt a service according to specific user behaviour and assumed requirements. Implicit personalisation has potential, but there are a lot of challenges that must be addressed. First, implicit personalisation requires that the user behaviour is recorded and stored. Second, this information must be processed and analysed. According to this analysis,

adaptation of the service behaviour can be performed. This analysis is difficult because user behaviour often can seem irrational. In addition to these challenges, privacy must be ensured of the recorded information of user behaviour. Direct marketing through SMS messages based on collected information about user behaviour has been seen as a powerful personalisation feature. A Personalization Consortium exists[1] that addresses personalised electronic marketing for the WWW. However, such personalisation does not necessarily fit with the fundamental idea that personalisation should be applied to *improve* the user's experience of a service. Massive reception of information for marketing purposes might easily become an annoyance. Pop up windows in browsers when surfing on the WWW is a good example. Similarly, SMS can be used to send directly targeted advertisements to cellular-phones and such spam has also become a problem [11].

Table 1 summarises personalisation types and whether the features are personalised explicit/implicit and what the motivation usually is for personalisation of each type.

	Explicit/Implicit	Motivation
Look & Feel	Explicit	Social Req.
Service Portfolio	Both	Work
Individual Service	Both	Social Req./Work
Content	Explicit	Work
Composition	Explicit	Work

Table 1: Classification of Personalisation

3.   Realising Personalisation

The mobility requirements combined with personalisation yields greater challenges than personalisation of stationary standalone services. To create a *mobile* service that is personalised, there are stricter requirements to service continuity and thus to the availability and accessibility of all components of the service at all times; even if this means across domains, networks and devices that are separated by security measurements (e.g. firewalls).

3.1   Challenges with Support for Personalisation

The personalisation process can be abstracted into the following three steps:

1. How can personal settings/information be gathered?
2. How are these settings/information stored in a persistent and ubiquitously available manner?

---

3. How can the changes always be reflected in the service rendering process?

Personalisation means that something is *changed* in/with a service, and this change *follows* the service at all time. If a user has personalised a service, it should stay personalised until the service contract is terminated or the user applies other changes that invalidate the previous settings. Thus, three important challenges with personalisation are:

The answers to these questions will help us with the design of the architecture for personalised mobile services.

3.2   Mechanisms to Support Personalisation

3.2.1   Gathering Personal Settings/Information

As already said, personalisation can be performed implicit or explicit. In general, an implicit personalisation process requires a component in the system that monitors user behaviour and makes changes to the profiles according to this behaviour. This is a common model used for personalised direct marketing on the WWW, but also by Microsoft products (e.g. MS Word) to make often used functions easily available in the menus.

On the other hand, if personalisation is to be explicit, more responsibility is transferred to the user application as specialised functionality, i.e., a user interface must exist that allows users to insert, modify and apply personal settings/information to the services.

Depending on which of the personalisation categories listed in section II.B is to be supported, this process of gathering settings will have quite different solutions.

3.2.2   Availability of User Profile

User profiles are used to store settings that define how services should behave for the users, i.e., they contain the settings that are applied to services to make them personalised. A user profile for a specific service is called the User Service Profile. Without the user or service having access to these profiles at all times, it will not be possible to render personalised services. As it will be required that services are available across networks, domains and devices, one challenge is to ensure access to these profiles while moving around between networks, domains and devices [12] [13].

In distributed computing, it is common to hide underlying functionality by applying transparencies [14]. For user profiles, one such transparency that should be investigated further is the replication transparency. Since profiles must be available across various networks, domains and devices, it will not be possible to realise this without creating replications of the data. A generic architectural model for management of replicated data can be found in [14]. User profiles provided as XML Web Services is considered in [15]

and XML Web Services in general are discussed in [16].

3.2.3 Availability of Device Profile

A user might have at disposition several different terminals, and thus to provide a personalised service, services must be able to adapt to each of these devices. The Device Independence Working Group at WWW Consortium states the need for *"Access to a Unified Web from Any Device in Any Context by Anyone"*. More generally, this should concern all services, web-based or not. Also, the device must be considered in conjunction with each user. Composite Capability/Preference Profiles (CC/PP) is a W3C Recommendation for describing device capabilities and user preferences [17]. A device profile states the characteristics of a terminal, and a Device Service Profile states how a service should be rendered on a specific device. Ubiquitous access to these profiles is important to ensure delivery of personalised services, and the challenge is similar to that of the User Profile.

Since availability of all profiles must be ensured across several types of boundaries they should be realised using platform independent technology. XML Web Services is therefore a good candidate for creating, managing and applying profiles, and the same concepts could be used as in the User Profile XML Web Service [15].

3.2.4 Availability of Personalised Content

Personalised content are documents that can be consumed or produced by user applications, and are often the result of service usage. Without ubiquitous access to such content, services are of less value to the consumer. A service on a stationary computer could for example be used to create a document that is stored in a home network. If this document is needed to render a service in a distant location (e.g. on a cellular phone), the accessibility of this document is crucial. No access means denial of service. A complete service architecture for personalised mobile services must thus ensure the accessibility of personalised content at all times.

3.2.5 Personalised Service Composition

By combining discrete service components, as described in Section II.B, it is possible to provide tailored services to consumers. Today, composition of services is restricted mostly to service developers and providers. By simplifying the interfaces and steps to combine service components, the responsibility and possibility of doing this can be moved to the end-user. The role of service developers and providers can thus in part be transferred to end-users.

3.2.6 Privacy

Privacy is always an issue in distributed computing, where personal content and personal information is travelling across various wireless or wired networks. With increased personalised aspects in services, the privacy should continue to be considered in every architecture and service. However, since security and privacy are advanced topics of their own, and due to space limitations, these issues are not considered in the proposed architectures throughout this paper.

3.2.7 Service Architecture Proposal

Figure 1 shows a general architecture for enhanced personalisation of mobile services. In particular, the new components are the *Personalisation Functions* on the client side, the *Monitor*, the *Service Composition Module* and the *Personalised Content Manager* on the server side. Each of the components on the server side has their own storage space where relevant data can be stored.
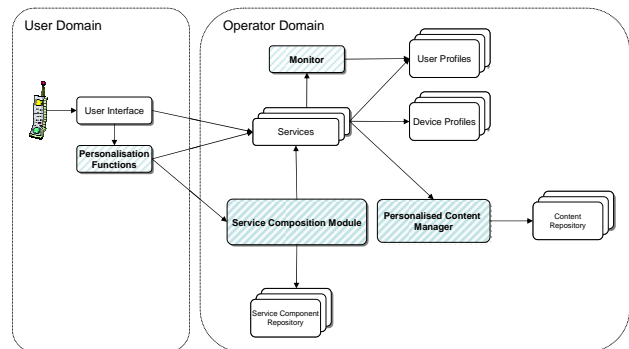

Figure 2: A general architecture for service personalisation

The monitor, together with the services themselves, manages the user and device profiles. The service composition module manages and uses a service component repository and the personalised content manager uses and stores content in a content repository.

The user profiles, device profiles and content repository have the same requirements to availability, and the support for these requirements are abstracted and hidden by the respective management components (i.e., profile manager and personalised content manager).

The service composition module has two tasks. First, it must be able to support the design phase of a personal service composition, and second, it must also support the runtime phase of a personal service composition. The design phase must typically ensure that only compatible components are joined, whereas the runtime phase must control the flow of data between components in the composition as well as return the resulting service to the client through the service access point (called service in Figure 2).

3.3 Personalisation with XML Web Services

Why should XML Web Services technology be used in personalisation of mobile services? The motivation for

using it with profiles has already been suggested; the platform independence is important. XML Web Services have well-defined interfaces (syntax), and their behaviour (semantics) is specified as part of the information provided in Web Services registries. Thus, XML Web Services can support compatibility matching between components that should be part of a service composition.

In its basic form, an architecture for personalisation will contain an end user device (e.g. cellular phone or PDA), several service components [12] and profiles for handling the specifics of personalisation. These are the requirements for a platform that supports the common personalisation concepts. However, to also support the extended types of personalisation (personalisation by content and personalisation by service composition), additional components must be introduced in the architecture. These components are the personalised content manager and the service composition components.

Figure 3 displays an architecture for personalised service composition. However, the user and device profiles are left out and should be considered separately. So also with the personalised content manager; the focus in the proposed architecture is on personalised service composition. In this architecture, Business Process Execution Language for Web Services (BPEL4WS) is used as an example technology for managing the process of combining Web Service (WS) components into a new WS; the personalised WS. The composition of a new personalised service is implemented as follows:

1. A user accesses the User Interface in the Composition Module and selects service components (exposed as XML Web Services) to fulfil the needed service
2. The WS Generator in the Composition Module generates a BPEL4WS process by orchestration of the selected XML Web Service components. The resulting service is described and exposed as a new XML Web Service.
3. The personalised service is now deployed and exposed to the user through a Portlet or another type of user interface

There are several challenges with this solution. First, there must be a User Interface which makes it possible for the user to select WS components to combine. Second, in the definition of a WS component A, information must be included that can be used to decide if it can be combined with another WS component B (i.e., do they have compatible inputs and outputs). Third, how can deployment of the generated WS be put through. Fourth, a Graphical User Interface (GUI) for the generated WS must be created. Such user interfaces must also take into account the various devices that the user might use to access the service. These challenges are now discussed.

3.3.1 Service Composition User Interface

For a user to be able to compose services, a user interface must exist that allows such composition. The interface must allow a user to:

1. Select *n* services
2. Decide if and how these could be interconnected
3. Generate composite Web Service

3.3.2 WS input/output compatibility

For any dynamic service composition, the components that are orchestrated must be *compatibility matched*, both on a syntactic and semantic level. First, and most importantly, the syntax of the outputs of one component must match the syntax of the expected input of the other component. For the semantic part, it must be ascertained that combining any two components makes sense, i.e., will information of type A have any value to a component B. Can the information be utilised to provide valuable output from B?

A framework for validating both the syntax and semantics of all components that can be part of service composition must exist to ensure WS compatibility in personalised service composition.

3.3.3 Deployment of personalised WS

When the XML documents (the orchestration schema [18]) that describe the orchestration of a service have been created, the service must be deployed in order to be available for access by the user. Usually, this task is performed by service providers or developers, but it is now the task of the user. The deployment should therefore be semi-automatic after necessary orchestration documents have been created. Also, validation of the composed service should be part of the deployment process.

3.3.4 Graphical User Interface for personalised WS

Having generated a Web Service and published this (e.g. through an UDDI), a user interface that the user can interact with is needed. In the architecture in Figure 3, the generation of this interface is performed by the composition module. To make this fully automatic is not a trivial task. For each input required by the user and for each output that should be displayed by the service, a set of widgets must be generated. One solution might be that each discrete Web Service component includes the specification of their own UI, and that the combination of this specification from the first and the last Web Service in the chain of executing Web Services are used to generate the final UI. This is, however, subject to future research.
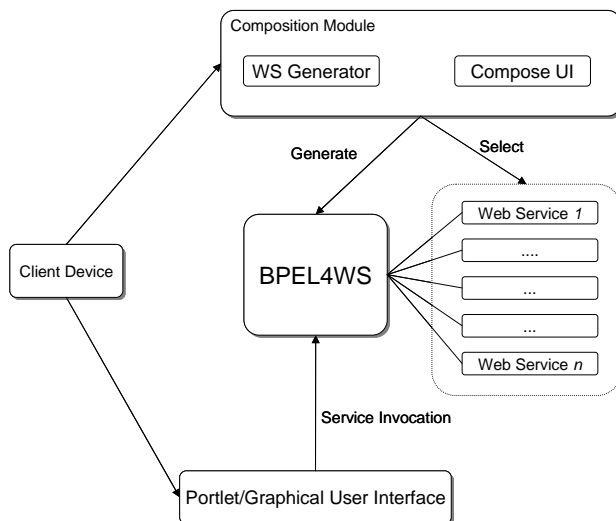
Figure 3: Personalised Service Composition architecture with BPEL4WS

3.3.5 Summary

BPEL4WS is a technology initially developed for service composition of advanced business processes, and as such it might be too complex to be suited for personalised service composition. A light-weight version capable of handling lots of users simultaneously and providing simple user interfaces for easy service composition should be further investigated, designed and implemented.

4. Conclusion and Future Work

Even though personalisation has been recognised as a powerful feature for improving mobile services, and thus potentially reducing churn by a network operator, the concepts and solutions are far from settled. Until now, personalisation has mostly been to accommodate social requirements, and this is popular among teenagers. However, personalisation to facilitate work, and especially collaborative work, has a great potential for growth. This paper highlights the ground-pillars of personalisation, and gives an overview of issues that must be considered to leverage future technologies that can support more advanced personalisation. It is emphasised that not only must management of User and Device Profiles be further studied, but also the management of products from service use and the opportunity for users to compose their own services at run-time should be considered in depth. Existing technologies for service composition are not suited for end-user composition, mainly because they are too complex. Thus the study of potential solutions to these challenges is important future work. The study of required mechanisms provided in this paper could be used as input to future studies.

References

[1] Ibrahim Cingil, et. al., A Broader Approach to Personalization, Communications of the ACM, Vol. 43, No. 9, August 2000

[2] Joseph Kramer, et. al., A User-Centered Design Approach to Personalization, Communications of the ACM, Vol. 43, No. 8, August 2000

[3] Jakob Nielsen, Alertbox for October 4, 1998, http://www.useit.com/alertbox/981004.html

[4] Nicole Harris, Personalisation Literature Review, 14th June 2002, available online at: http://www.angel.ac.uk/public-files/pdf/personalisation.pdf

[5] John O'Looney, The Future of Public-Sector Internet Service: Pt. 1, A national survey reveals the state of government participation on the web, September 2001, available online at: http://www.govtech.net/magazine/story.print.php?id=5806

[6] Jan Blom, Why do we personalise?, 4th Human Centred Technology Postgraduate Workshop, 3rd & 4th October 2000

[7] Schahram Dustdar, Service Composition, Panel Discussion, The First International Conference on Service Oriented Computing, Trento, Italy, 15-18 December, 2003, available online at:http://www.unitn.it/convegni/download/icsoc03/panel/P_P2_Dustdar.pdf

[8] Milner, R., Communication and Concurrency, Prentice Hall International Series on Computer Science, 1989, ISBN 0-13-114984-9

[9] Mennie, D., and Pagurek, B., "An Architecture to Support Dynamic Composition of Service Components", Proceedings of the 5th International Workshop on Component-Oriented Programming (WCOP 2000)

[10] Shuk Ying and Sai Ho Kwok, The Attraction of Personalized Service for Users in Mobile Commerce: An Empirical Study, ACM SIGecom Exchanges, Vol.3, No.4, January 2003, Pages 10-18

[11] ITU, ITU WSIS Thematic Meeting on Countering Spam, Chairman's Report, Geneva, Switzerland, 7-9 July 2004

[12] Jørstad, I., van Do, T., Dustdar, S. (2004). An analysis of service continuity in mobile services. 2nd International Workshop on Distributed and Mobile collaboration (DMC), WETICE conference, 14-16 June, Modena, Italy, 2004, IEEE Computer Society Press

[13] Jørstad, I., Dustdar, S., van Do, T. (2004). *Service Continuity and Personalisation in Future Mobile Services.* 10th International IFIP Workshop on Advances in Fixed and Mobile Networks (EUNICE), Tampere, Finland, 14-16 June 2004

[14] George Coulouris et. al, Distributed Systems: Concepts and Design, 3rd Edition, ISBN 0-201-619-180, Addison-Wesley, Pearson Education, 2001

[15] Hartvigsen Anne Marie & Jønvik Tore van & Do van Than: Offering User Profile as an XML Web Service - Databases and Information Systems II - The Fifth International Baltic conference on DB and IS, Tallinn, Estonia, June 3-6, 2002 - Selected Papers - Kluwer Academic Publishers ISBN 1-4020-1038-9

[16] Erik Vanem & Do van Thanh: What is an XML web service? - Telektronikk Volume 98 No 4 - 2002 ISSN 0085-7130

[17] WWW Consortium, Composite Capability/Preference Profiles (CC/PP): Structure and vocabularies 1.0, W3C Recommendation, January 15th, 2004, available online at: http://www.w3.org/TR/2004/REC-CCPP-struct-vocab-20040115/

[18] Gustavo Alonso, et.al., Web Services: Concepts, Architectures and Applications, Springer-Verlag, 2004, ISBN 3-540-44008-9