Personalisation of Next Generation Mobile Services

Ivar Jørstad[1], Do van Thanh[2] and Schahram Dustdar[3]

[1] Norwegian University of Science and Technology, Dept. of Telematics, O.S. Bragstads plass 2E, N-7491 Trondheim, Norway
ivar@ongx.org, http://ongx.org
[2] Telenor R&D, Snarøyveien 30 N-1331 Fornebu, Norway
thanh-van.do@telenor.com http://www.item.ntnu.no/~thanhvan
[3] Vienna University of Technology, Distributed Systems Group (DSG), Information Systems Institute A-1040 Wien, Argentinierstrasse 8/184-1, Austria
dustdar@infosys.tuwien.ac.at
http://www.infosys.tuwien.ac.at/Staff/sd/

Abstract. As communication technologies are becoming more and more advanced, the opportunities to deliver improved, user friendly services using these technologies are increasing. One of the possible ways to improve the services is to enable personalisation. However, to enable personalisation it is not sufficient to consider the service on its own as it has been the case until now. It is necessary to consider personalisation as a higher-layer function, which should span different services, be it different service instances or different service implementations. Personalisation should also span different terminal platforms and different network technologies. It is therefore appropriate to consider personalisation as a function in the higher-layers of the service platforms, above or integrated with the existing middleware. This paper provides a precise definition of personalisation and clarifies how it will be possible to build advanced mobile services in the future which make extensive use of the personalisation function to improve the perceived value and quality of these services.

1   Introduction

Personalisation of services is to adapt services to fit the needs and preferences of a user or a group of users. There exist many different definitions of personalisation. In [1], personalisation is defined as:

"Personalisation of a service is the ability to allow a user U to modify or produce, a service A such that it fits user U's particular needs in terms of presentation and functionality, and after such personalisation, all subsequent service rendering of service A for user U will be conformed to the performed modification."

Although the definition states that personalisation is done by the user, most of the tasks are done by either the service itself or the service platform. In fact, it only

stresses that the user should be the one in charge and initiating the personalisation process in the first place.

Personalisation is important in today's service-oriented society, and has proven to be crucial for the acceptance of services provided by the Internet and mobile telecommunication networks (illustrated by the success of personalised ring tones, logos, etc.). In [2], the motivation for personalisation is described and two important categories of personalisation are identified: *personalisation to facilitate work* and *personalisation to accommodate social requirements*.

In the first category, services are adapted to increase the efficiency, e.g. to minimize the time spent on repetitive and similar work tasks. The adaptation can aim at accommodating physical differences of the users like weaksightedness, disabilities, etc.

In the second category, services are adapted to enhance the social experience. For example, youngsters, by changing the appearance and behavior of a cellular phone (ring tone, logos etc.) want to express their identity/personality.

To enable service developers and providers of both the Internet and mobile telecommunication networks to support personalisation, adequate middleware and service platforms must be present. They act as a fundament and catalyst for increasing the number of personalised services.

Until recently, each specific communication technology defined its own sphere for service delivery. The GSM network has provided users with GSM voice telephony and SMS messaging services. The Internet has provided users with e-mail and the World Wide Web (WWW). Bluetooth technology has provided users with short-range communication services, interconnecting personal devices like personal computers and cellular phones. On the network side there has been a slight integration over the years; for example, GSM is extended with GPRS, which provided access to the Internet from GSM based devices.

Today, the integration is moving towards the end-user terminal. Devices are becoming multi-modal, thus allowing access to different service delivery networks by switching between network access points and using different radio access technologies. This is enabled by integration of several radio access technologies in each end-user terminal. Some terminals now include GSM, WLAN and Bluetooth radio access modules. Efforts are made to allow hand-over and roaming facilities across these radio access technologies, e.g. as envisioned by the Unlicensed Mobile Access (UMA) initiative [3]. This could allow access to GSM specific services even when using WLAN or Bluetooth as the radio access method.

With these developments in the terminals, a lot of new opportunities arise. In particular, advanced personalisation of services is becoming feasible. However, this is not enabled by itself. Even though networks and service platforms are being tighter integrated, services currently employ their own mechanisms for personalisation, thus prohibiting personalisation across service implementations.

Thoughtful design and implementation of proper functions in the upper-layers of the service delivery platforms is hence required. This paper provides an extensive discussion of what personalisation is, how it is related to mobile services, what functions it is dependent on and how it can be realised in future service delivery platforms. Section 2 presents related works. Section 3 provides a discussion of future mobile

services and the requirement of personalisation. Section 4 proceeds with a conceptualisation of personalisation and the elaboration of required ontologies.

2 Related Works

This paper builds on work on personalisation in [1] and [4] and provides extensions. In addition, there are efforts at different working groups at ETSI that will be successively summarized.

2.1 ETSI User Profile Management

ETSI has released some guidelines for User Profile Management [5]. These guidelines cover a lot of ground, among others the concepts of user profiles, stakeholders and roles, rules and maintenance of profiles. The introduction states the following: *"The present document focuses on presenting guidelines to service providers and manufacturers in shaping their product requirements in ways to maximize human and social benefit."* The guidelines also point to the importance of user profile management for the uptake and success of new and advanced communication services.

2.2 ETSI Generic User Profile (GUP)

ETSI has also released a set of technical specifications [6][7][8] which define a Generic User Profile (GUP) for the 3GPP mobile system. The specifications state that: *"The objective of specifying the 3GPP GUP is to provide a means to enable harmonised usage of the user-related information originating from different domains."*

The specifications recognize that user profile data should be shared between different stakeholders to facilitate the following:

- *User preference management*
- *User service customization*
- Terminal capability management
- *User information sharing*
- Profile key access

Of the above listed areas, the *italicized* items will in particular be considered in this paper, since (as will be shown) they are of importance to the personalisation of mobile services.

The ETSI guidelines and specifications are focusing on important areas, although they tend to be very telecom centric. This paper contributes to providing supplementary knowledge around both the general concepts of personalisation as well as towards the realization of personalisation in future service platforms.

3 Future Mobile Services and Personalisation

This section gives an introduction showing how personalisation will be important for future mobile services. To be able to elaborate on this, it is first necessary to study the characteristics of existing mobile services and how future mobile services could be composed, deployed and consumed.

3.1 Future Mobile Services

Being mobile relates to the ability to move. A mobile service however, does not necessarily mean that the service moves. Instead, it is a service which can be accessed when the user moves. It is possible to move parts of the service to achieve this, but the more common way to implement a mobile service is to dynamically change the *service composition* according to the user movements. This is best illustrated with the most common mobile service today; mobile voice telephony (e.g. GSM). When a user roams between GSM network operators, the service will change its internal structure (its service composition). As shown in **Fig. 1**, the voice telephony service, originally composed of components in the Home Location Register, Authentication Centre (AuC) and Mobile Switching Centre (MSC) will be realized by a VLR and a new MSC when the user is moving.
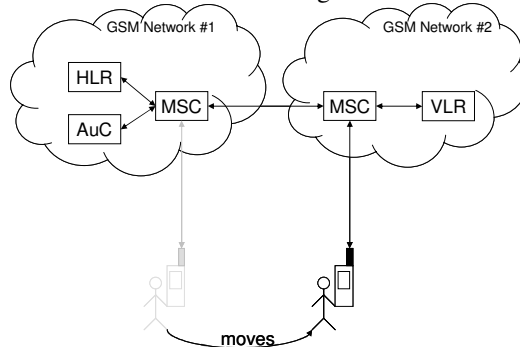

Fig. 1. When roaming in GSM, the service composition is changed (abbrev: HLR – Home Location Register, AuC – Authentication Centre, MSC – Mobile services Switching Centre, VLR – Visitor Location Register)

Future mobile services should build on and extend the architectural concepts of GSM, in order to allow maximum mobility of the user. In particular, the realisation of mobile services in GSM exhibits the following characteristics:

1. The composite service (GSM voice telephony) is realised by service components partly on a user device and partly in a network
2. The same composite service can be realised by different service components (different instances) for different users and different devices

3. The same service can be realised by service components developed by different manufacturers (different implementations) and located in different service provider locations
4. The composite service in GSM changes its internal composition to accommodate user movements
5. Movement between terminals, while still receiving the same services, is supported by the use of a personal token; the Subscriber Identity Module (SIM)

Future mobile services can in addition benefit from adopting Service-Oriented Architecture (SOA) [9] concepts as proposed in [10][11]. SOA supports characteristics like loose-coupling, dynamic service discovery and composition.

The resulting service architecture can allow:

- Users to move geographically over distances using a single terminal (which involves different network access points)
- User to keep service access when the user terminal switches between network access points of different type, e.g. due to using different radio access technologies
- Users to move between different terminals (e.g. between cellular phone and PC) while still receiving the same service

3.2 Mobile Services Decomposition

To simplify and make the understanding easier, it is possible to abstract a mobile service into one coherent unit, which is a service in its own right, as defined by for example [12]. However, to be able to carry out an in-depth study of how future mobile services can be engineered to support personalisation, it is necessary to consider the basic elements (building blocks), which any mobile service should consist of. **Fig. 2** illustrates the composition of a generic mobile service, as introduced in [13]. The motivation for this decomposition is as follows:

First, every service is realized by a logic, i.e. program code that accepts input, processes and provide some output (often also referred to as application logic or software). The service logic is without doubt the first basic element of a mobile service.

Second, to allow a service to change of service logic without interruption, it is necessary that the replacing logic received the internal state data of the previous service logic. The second basic element of a mobile service is therefore the service data.

Third, many services are related to the storage, transfer, processing and rendering of information or content such as voice, video, text, etc. to the users. Such a content should also persist after the termination of a service session and can be reused by later session. The third basic element is therefore the service content. The difference between service data and service content is that service data is considered transient, thus it has relevance in one service session, whereas service content has relevance across service sessions. Note however that the definition of session can be slightly different for different types of services. Consider the following example. A session of a Web Browser lasts from it is started by double-clicking the application icon, until the browser is closed; surfing different web-pages is not considered as different sessions.

Thus, service data in a browser can for example be the history of URLs for the current session, whereas bookmarks are considered service content. Most browsers keep the history persistent also, so the borderline between service data and service content can in many cases be difficult to define. The distinction may be that the service content is of interest to the user while the service data is only interesting for the service execution.

Fourth, services tend to include preferences that can be changed by the user. A lot of the information elements that personalisation depends on, are parts of this component. This component will be further studied in the next sections. Thus, service profile is the fourth and last component of the generic mobile service. To summarise, the basic components of a generic mobile service are:

ServiceLogic – Application, program code, software
ServiceData – Transient data (e.g. input, output and internal state)
ServiceContent – Persistent data (e.g. documents)
ServiceProfile – Preferences (e.g. service settings like font colors, how the service should behave)
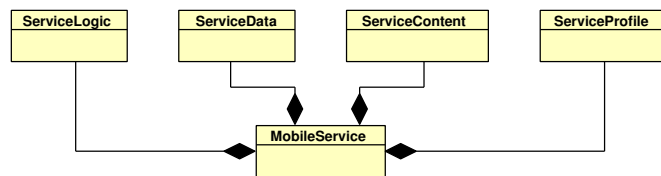

Fig. 2. The composition of a generic mobile service

3.3 Personalisation Requirements

The focus of personalisation today is mostly towards services on the World Wide Web (WWW). When a service on the WWW is personalised, it stays personalised even when the user accesses the service from different devices (e.g. two different PCs), since the configuration related to personalisation is stored on the service provider side. With regards to personalisation of desktop services (e.g. applications in MS Windows), the configuration related to personalisation is stored locally on the specific device.

Personalisation of future mobile services must cope with both scenarios. It is not enough to consider personalisation of WWW based services only, because even these are dependent on locally stored parameters (e.g. browser configuration and bookmarks).

As illustrated by sub-section 3.1, two of the challenges with personalisation will be to allow *cross-device* and *cross-network access* to personalisation information. In addition, as proposed by guideline 4.1.4.a in [5], it should be possible to allow *cross-service access* to personalisation information. This means that conceptually different services should be able to reuse generic preferences. These requirements, and additional requirements, will be discussed in the succeeding sections.

Personalisation is realised using the information contained by the Personalisation Information Space [4] as illustrated in **Fig. 3** . The personalisation of a service is centered around the following two issues:

- o The Personalisation Information that defines the personalisation
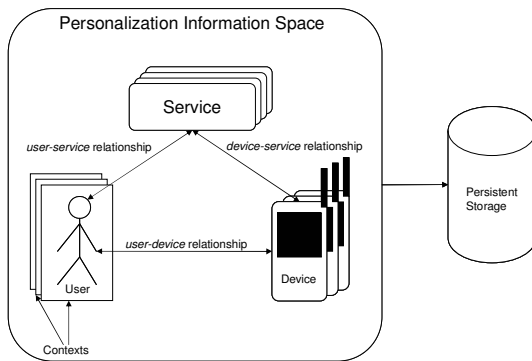- o The functionality to apply personalisation

Fig. 3. The Personalisation Information Space

Cross-device personalisation - To enable personalisation across devices, of both equivalent and different types, is a requirement. However, cross-device personalisation is not trivial. As **Fig. 4** illustrates, there are several possibilities that must be considered when cross-device personalisation is to be supported.
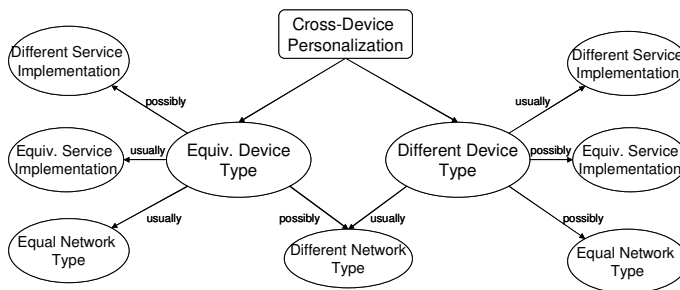

Fig. 4. Possibilities to consider when supporting cross-device personalisation

4   A Conceptual View of Personalisation

Since personalisation shall be possible across a range of various devices, networks and services, it is necessary to first develop conceptual and abstract models that are independent of the underlying technologies. Section 4.1 develops a conceptual/high-level meta-data model of the information that enables personalisation. This model

should be applicable for all types of mobile services. Section 4.2 conceptualises personalisation as a relationship between the user and the service, to expose the requirements posed in particular by cross-service personalisation.

4.1 Personalisation Information

This section elaborates on the different parts of the personalisation information. A UML class diagram illustrating the various components of the personalisation information is shown in **Fig. 5**.

Personalisation Information – This component represents the aggregate of all personalisation information. Ideally, this component should be provided as a shared component among as many service providers in as many service domains as possible.

User Personalisation Information – This component contains all personalisation information for one user.
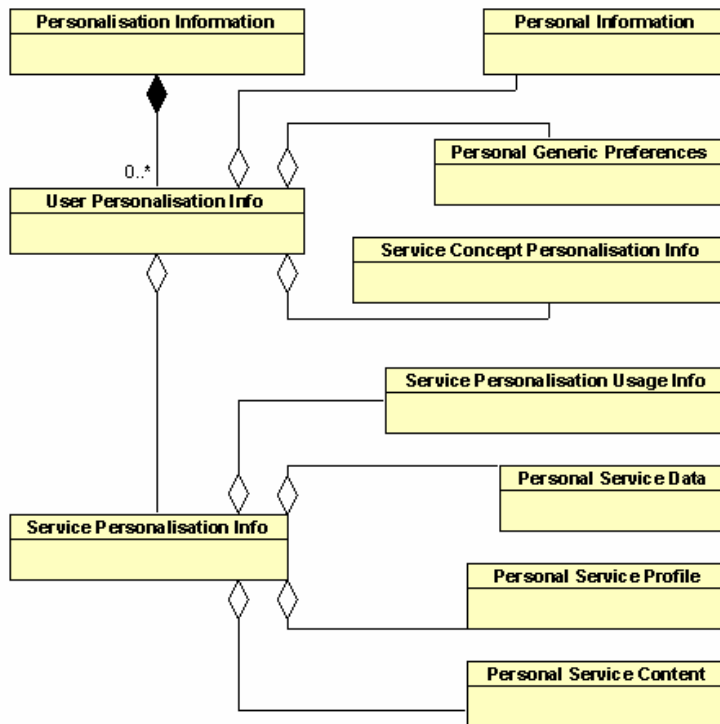

Fig. 5. The components of the Personalisation Information

Personal Information – This component contains personal information for a user. Examples of such information include addresses, phone numbers, credit card information etc. This information is in many cases unique, and more or less static, for each user.

Personal Generic Preferences – This component contains preferences that are generic to all services and service types/concepts. This could for example be font size settings and color selections.

Service Concept Personalisation Information – This component contains preferences that are generic to all services of a given type (henceforth the term service concept is used instead of service type). Thus, personalisation information that is common among several service implementations is moved out of the Personal Service Profile (where it would usually reside, according to **Fig. 2**) and into this component instead.

Service Personalisation Information – This component is the aggregate of all personalisation information that are specific to one service.

Service Personalisation Usage Information – This component contains information about the usage of a specific service.

Personal Service Data – This component contains service data that are related to a specific service.

Personal Service Content – This component contains service content that is related to a specific service.

Personal Service Profile – This component contains service profile that is related to a specific service.

4.2   Service Conceptualisation and the Personalisation Relationship

Fig. 6 displays a conceptualization of a service, and includes some terms that are used in the next sections.
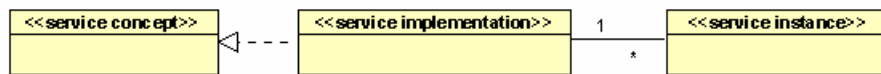

Fig. 6. A conceptualization of a service

The following meaning and relationship among these terms are assumed:

Service Concept – An abstract idea of a service; e.g. the idea of a WWW browser or a word processor.

Service Implementation – A realization of a service concept; e.g. Internet Explorer is a realization of the service concept WWW browser. A service implementation can realize one or more service concepts (for example, a WWW browser can typically open local text documents, and the WWW browser Opera is also an e-mail client).

Service Instance – The unique instance of a service implementation in a specific location; e.g. the installation of Internet Explorer on a PC.

The primary enablers of personalisation are the elements that contain information linking *users* to *services*. These are elements of the Personalisation Information Space as previously defined in [4]. In **Fig. 7**(a.), there is one personalisation relationship for each service instance. Thus, this is defined as the concept of *Local Service Personalisation* (to indicate that the personalisation is *local* to the specific *service instance*):

Local Service Personalisation – Personalisation is constrained/limited to cope with each individual service instance, and thus also with each service implementation.
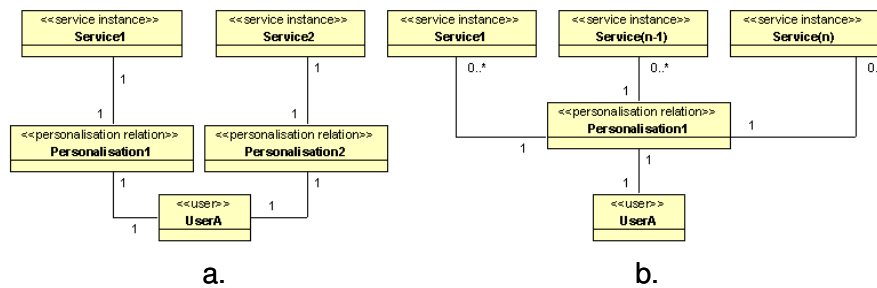

Fig. 7. Local Service Personalisation

In **Fig. 7**(b.), there is one personalisation relationship shared by a number of service instances. This is the concept of Global Service Personalisation.

Global Service Personalisation – In Global Service Personalisation, several service instances can be personalised by the same personalisation information.

However, the two previous models illustrating the personalisation relationship do not consider the service implementation, or rather *different* service implementations. **Fig. 8**(a.) shows the concept of different service instances of the same implementation being personalised, while **Fig. 8**(b.) shows the case where different service instances of different service implementations are personalised through the same personalisation relationship.

Fig. 8(b.) shows the ultimate goal, where personalisation is independent of both service instance and service implementations. Note also that both implementations in **Fig. 8**(b.) are realizations of the same concept; for the most cases, this is a requirement, because it seldom makes sense to use personalisation information from a service of one concept, in another service realising another concept (e.g. using an MS Word document with an mp3-player would in most cases yield noise). However, generic preferences are possible, so it is also necessary to consider a Global Cross-Service Concept Personalisation, as depicted in **Fig. 9**.
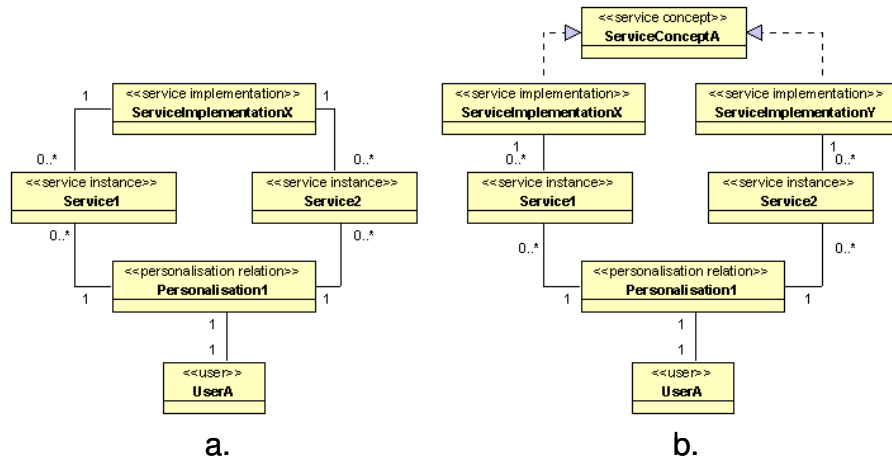
Fig. 8. Global Service Personalisation (a) and Global Cross-Service Personalisation (b)
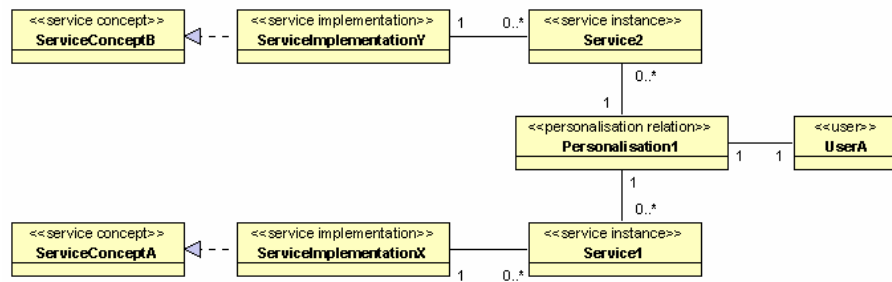

Fig. 9. Global Cross-Service Type Personalisation

The condition for Global Cross-Service Type Personalisation is the ability to compare and conclude about the equivalence or difference between the service implementations. More specifically, it is necessary to find:

- How closely related are the concepts they realize (semantics)
- How different are the implementions (syntax)

These issues will be considered in the next section.

4.3 Organisation of Personalisation Information

To enable the comparison between service implementation and personalisation information, it is necessary to define an ontology specifying the organizational structure between services. It is necessary to define relationship between services and the rules to navigate in the structure. For example, given two services it should be always possible determine the relation between them (equivalent, different, subset, etc.)

Fig. 10 shows how different personalisation information elements (represented as stereotyped *<<PIE>>* in the UML class diagrams) are associated with a specific

service implementation through a service concept. It should thus be possible for a service implementation to both inherit a specification from the service concept, and to provide an additional specification by creating an association directly to a personalisation information element
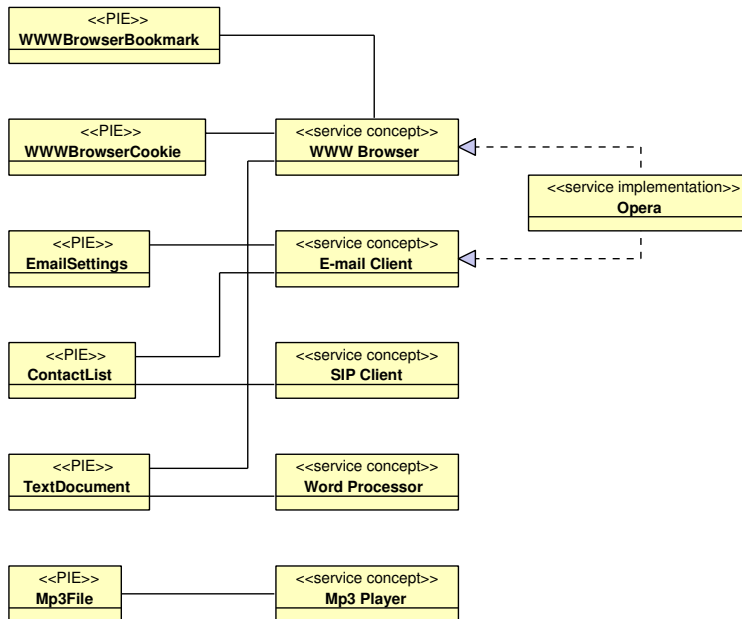

Fig. 10. Personalisation Information Elements (PIE) associated to service implementations through service concepts

4.4 OWL Ontology Specification

The model described in the previous section is only a visual representation of the personalisation information elements and their relationships to service concepts and implementations. To be machine processable, the visual model must be transformed into a textual representation.

The OWL Web Ontology Language [14] is based on RDF [15], and is a language for defining and instantiating Web ontologies. Below is the specification of the personalisation information ontology in abstract OWL syntax [16][17]. Only one service concept is covered (the WWW browser) in this example ontology.

```
Ontology(
Annotation(rdfs:label "Personalisation Ontology")
Annotation(owl:imports http://www.ongx.org/service_1)
Annotation(owl:imports http://www.ongx.org/service_2)
…
Annotation(owl:imports http://www.ongx.org/service_n)
```

```
ObjectProperty(personalised-by inverseof(personalises))
ObjectProperty(personalises domain(serviceConcept))
ObjectProperty(realised-by inverseof(realises))
ObjectProperty(realises domain(serviceConcept))
Class(serviceImplementation partial annotation(rdfs:comment "A specific
service implementation"))
Class(serviceConcept partial annotation(rdfs:comment "The abstract con-
cept of a service"))
Class(piElement partial annotation(rdfs:comment "A personalisation in-
formation element"))
Class(piSet complete annotation(rdfs:comment "The set of all personal-
isation information elements") piElement)
Class(WWWBrowserBookmark partial piElement)
Class(WWWBrowserCookie partial piElement)
Class(TextDocument partial piElement)
Class(Mp3File partial piElement)
Class(WWWBrowser partial serviceConcept
restriction(personalised-by    WWWBrowserBookmark    WWWBrowserCookie
TextDocument))
Class(Opera partial serviceImplementation
restriction(realises WWWBrowser))
)
```

The ontology starts by including already existing ontologies (using *owl:imports*). The ontology should as such be expandable, and could in theory include ontologies developed by various service providers.

The challenge for personalisation is now to populate the ontology to contain information about all available mobile services. This can either be done by careful analysis of already existing services, or it can be left for developers of new services. For storage, the most appropriate will probably be to have a centralised registry where service developers can submit their additions to the ontologies, either bare linking them (i.e., using the *imports* feature of OWL) or inserting the additions into the top-level ontology.

With the ontologies in place, there already exist several reasoning engines which can be used for processing the OWL specifications. One of the remaining challenges now is for service implementations to be able to exchange personalisation information which is semantically the same (due to the use of ontologies) but which have different representations. The next step is thus to define the necessary functionality of the personalisation architecture, which will include syntax validation and transformation etc. Due to space limitations, this elaboration is not included in this paper.

5   Conclusion

This paper contributes to the advance of mobile services by providing a thorough analysis of service personalisation. An introduction to personalisation is provided, and the core concepts of personalisation are discussed. The personalisation information space is conceptualized, to provide a framework for personalisation of any future mobile service.Based on this conceptualization, an ontology for personalisation information elements is modelled. The modelling is done in UML, and then transformed to an OWL specification using the abstract syntax notation.

References

[1] Jørstad, I. & van Do, T. (2004). "Personalisation of Future Mobile Services". 9th Interational Conference on Intelligence in service delivery Networks (ICIN), Bordeaux, France, October 18-21, 2004

[2] Blom, J. (2000), "Personalization – A Taxonomy", Conference on Human Factors in Computing Systems (CHI), Hague, Netherlands, April 1-6, 2000, ISBN: 1-58113-248-4

[3] Unlicensed Mobile Access (UMA), "Unlicensed Mobile Access (UMA); Architecture (Stage 2) R1.0.0", Technical Specification, September 1, 2004, http://www.umatechnology.org/specifications/index.htm

[4] Jørstad, I., van Do, T. & Dustdar, S. (2005). "The Personalisation of Mobile Services", 1st IEEE International Conference on Wireless and Mobile Computing, Networing and Communications (WIMOB), Montreal, Canada, August 22-24, 2005, ISBN: 0-7803-9182-9

[5] ETSI. (2005). "ETSI EG 202 325 v0.0.11: Human Factors; User Profile Management". ETSI, 2005

[6] ETSI (2005). "TS 122 240 v6.5.0: Universal Mobile Telecommunications System (UMTS); Service requirements for 3GPP Generic User Profile (GUP); Stage 1", ETSI, 2005

[7] ETSI (2005). "TS 123 240 v6.7.0: Universal Mobile Telecommunications System (UMTS); Service requirements for 3GPP Generic User Profile (GUP); Stage 2", ETSI, 2005

[8] ETSI (2005). "TS 129 240 v6.0.0: Universal Mobile Telecommunications System (UMTS); Service requirements for 3GPP Generic User Profile (GUP); Stage 3", ETSI, 2005

[9] Papazoglou, M.P. and Georgakopoulos D. (2003). "Service-Oriented Computing". Communications of the ACM, Vol. 46, No. 10, October, 2003

[10] Jørstad, I., Dustdar, S., van Do, T. (2005). "Service-Oriented Architectures and Mobile Services", Ubiquitous Mobile Information and Collaboration Systems (UMICS2005), Porto, Portugal, 13-14 June 2005

[11] Jørstad, I., van Do, T. (2005). "A Service-Oriented Architecture Framework for Mobile Services", Advanced Industrial Conference on Telecommunications (AICT2005), Lisbon, Portugal, 18-22 July 2005, ISBN: 0-7995-2388-9

[12] Sassen, A. M. & MacMillan, C. (2005). "The service engineering area: An overview of its current state and a vision of its future", European Commission, Information Society and Media Directorate, Network and Communication Technologies, Software Technologies, 1 July 2005

[13] Jørstad, I., Dustdar, S., van Do, T. (2004). "Towards Service Continuity for Generic Mobile Services". The 2004 IFIP International Conference on Intelligence in Communication Systems (INTELLCOMM 04), Bangkok, Thailand, 23-26 November 2004. ISBN: 3-540-23893-X

[14] Smith, M.K., Welty, C. & McGuiness, D.L. (2004). "OWL Web Ontology Language", W3C Recommendation, February 10, 2004, http://www.w3.org/TR/owl-guide/

[15] Beckett, D. (ed.) (2004). "RDF/XML Syntax Specification (Revised)". W3C Recommendation, 10 February 2004, online: http://www.w3.org/TR/rdf-syntax-grammar/

[16] Antoniou, G. & Harmelen, van F. (2004). *A Semantic Web Primer*. MIT Press, 2004, ISBN: 0-262-01210-3

[17] Patel-Schneider, P.F., Hayes, P. & Horrocks, I. (2004). "OWL Web Ontology Language Semantics and Abstract Syntax", W3C Recommendation, February 10, 2004, http://www.w3.org/TR/2004/REC-owl-semantics-20040210/