A Reference Model for Service Oriented Middleware

Marco Autili, Mauro Caporuscio, Valérie Issarny

To cite this version:

Marco Autili, Mauro Caporuscio, Valérie Issarny. A Reference Model for Service Oriented Middleware. [Research Report] 2008. inria-00326479

HAL Id: inria-00326479
https://hal.inria.fr/inria-00326479

Submitted on 3 Oct 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Reference Model for Service Oriented Middleware[*]

Marco Autili[1], Mauro Caporuscio[2], and Valerie Issarny[2]

[1] Dipartimento di Informatica - Università degli Studi di L'Aquila, Italy
`marco.autili@di.univaq.it`
[2] INRIA Paris-Rocquencourt - 78153 Le Chesnay, France
{`mauro.caporuscio, valerie.issarny`}`@inria.fr`

Abstract. From the software engineering perspective, the notion of Service Oriented Architecture (SOA) has been receiving significant attention within the software design and development community. This attention has led to the proliferation of conflicting characterizations of SOA, resulting in an ambiguous understanding of SOA entities and relationships among them. To achieve a common understanding, OASIS and SeCSE propose reference models that introduce a comprehensive ontology for modeling software services around the well known service-oriented interaction pattern. However OASIS and SeCSE models abstract the actual interaction pattern runtime support, which is generally provided by a Service Oriented Middleware (SOM). In this paper we propose a reference model for architecting SOM solutions over next generation networking environment and evaluate it by designing a conforming lower-level model for the SOM developed for the PLASTIC project.

1 Introduction

Service Oriented Architecture (SOA) has been largely accepted as a well founded reference architectural style for an ever larger class of application domains, spanning from e-commerce to pervasive computing. SOA relies on the service-oriented computing paradigm that considers services as building blocks for developing distributed applications. Networked devices and their hosted applications are then abstracted as autonomous loosely coupled services that can be easily integrated into a network of services to create flexible and dynamic processes.

From the software engineering perspective, the notion of SOA has been receiving significant attention within the software design and development community. This attention has led to the proliferation of conflicting characterizations of SOA, resulting in an ambiguous understanding of what are the SOA entities (and the relationships among them) to be considered while developing service-oriented applications. Valuable work in the Literature, such as OASIS [14] and SeCSE [12], propose reference models for achieving a common understanding of

---

[*] This work is part of the IST PLASTIC project and has been funded by the European Commission, FP6 contract number 026955, http://www.ist-plastic.org/.

the SOA style. Specifically, being not tied to any standard, technology or other concrete implementation details, these models describe actors, roles, activities and entities, as well as relationships between them, by introducing an abstract comprehensive ontology for characterizing SOA. They further enable the development of specific reference and concrete service-oriented architectures using consistent standards and specifications. In particular, the SeCSE model complements the one of WSA [19] by clarifying the concept of service and service-related activities (e.g.,, publication, discovery, composition, and monitoring), as well as the relationships between service description, semantics and QoS. However, these models do not specify how the service-oriented interaction pattern should be supported at runtime.

Software engineering best-practices suggest the exploitation of a Service Oriented Middleware (SOM) to support the service-oriented interaction pattern through the provision of proper features for deploying, publishing, discovering and accessing services at runtime. In general, middleware are technologies aimed at supporting distributed applications by masking the distribution and heterogeneity of the execution and networking environment [9]. However, next generation networking environment, such as Beyond 3^{rd} Generation (B3G) networks [20], can be no longer considered as “passive” entities which only transport data between end-points and, as such, they cannot be masked by an homogeneous layer. Rather, they must be considered as “active” parties to be fully exploited for benefiting of the underlying network diversity that might change over time. B3G pursues the convergence of wireless telecommunication networks and wireless IP networks (e.g., UMTS, WiFi and Bluetooth) to provide connectivity from everywhere at anytime thanks to the availability of newest multi-radio devices like smart phones and PDA. Supporting SOA in such complex networking environment requires for a *B3G-oriented SOM* (B3G-SOM from now on), which does not homogenize in a systematic way the diversity and richness of the networking infrastructure. Rather, B3G network characteristics should be made available to and exploitable by the service layer depending on the specific service requirements.

Recently, in [15] the authors propose a layered architecture that conceptually separates SOA functional aspects, as well as elicit extra-functional cross-cutting concerns such as semantics and QoS. In particular they pose SOM at the bottom layer for realizing runtime SOA infrastructure. In [21], the author defines a pattern-based approach to SOMs’ conceptual design by combining well known concepts derived from existent technologies (e.g., Web services, coordination models, semantic Web) with SOA concepts. Still, these approaches do not consider the heterogeneity and richness of next generation B3G networks.

To the best of our knowledge, there is no attempt to define an abstract reference model that comprehensively takes into account the complexity of the B3G-networking environment, how it affects the service-oriented computing paradigm and, hence, the design of a SOM for such an environment. Being inspired by the work in [12,14,15,21], in this paper we propose a *B3G-SOM reference model* that brings together notions to be accounted while architecting middleware solutions for service-oriented applications over B3G. The model aims at establishing a common understanding of the support that should be provided by a generic B3G-SOM and can be considered as a facilitator for designing architectures of specific B3G-SOM solutions.

In order to elicit the main B3G related concepts to be included in our reference model, we consider already in place and under development valuable enabling technologies - i.e., IMS [1], UMA [2] and IEEE 802.21 [8] - that, at different levels and hence in different ways, tackle B3G related issues. We then filter out their commonalities and include into our model those notions we identify as necessary to deal with the B3G domain. We further consider principles and best-practices that, from a software engineering perspective, should be accounted while developing middleware-based applications [9]. However, our model is not to be understood as a holistic model, rather it is to be considered in conjunction with the models discussed above to bring together first-class concepts related to SOM with concepts related to the wide B3G and SOA domains.

The paper is organized as follows. Section 2 discusses motivations and goals. Our reference model is introduced in Section 3 and it is fully presented in Section 4. Section 5 presents then a conforming lower-level reference model for the B3G-SOM developed within the PLASTIC project. Finally, Section 6 concludes the paper by summarizing our contribution and discussing future perspectives.

2 Motivations and Goals

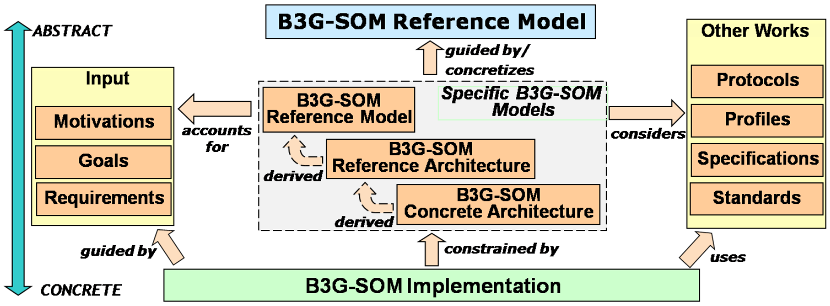

Fig. 1. Reference model exploitation

In this section we discuss motivations and goals for a reference model, and how it can be used for designing SOM solutions by also identifying the audience it is aimed at. The purpose of the proposed reference model is to bring together notions to be accounted while architecting middleware solutions for service-oriented applications over B3G networks. That is, the entities and relationships within our model (together with its rich textual description) should be intended as base guidelines to be considered while designing B3G-SOM architectures.

Figure 1 shows an example of how the architectural specification of a B3G-SOM might be guided by our reference model. A *concrete architecture* (the specific final solution) might be derived from a *reference architecture* (class of solutions) that, in turn, might derive from a *lower-level SOM reference model*. In fact, our model is a high-level model and independent from whatever specific platform and specific middleware solution. Hence, our model can be customized/instantiated into a lower-level B3G-SOM reference model that constitutes a richer and more concrete reference model by showing, for instance, architectural elements as well as provided features and deployment information. More than one "lower-level model" can be derived by incremental refining steps supported, e.g., by model driven engineering techniques. Still referring to Figure 1 both reference and concrete architectures, as well as lower-level models, should generally be designed by accounting for specific requirements, motivation and goals of the specific middleware to be architected, as well as considering specific standards, protocols, profiles, and specifications that can be (re)used (e.g., WSA standards). Finally, the *SOM implementation* should be constrained by all the artifacts from the lower-level model(s) to the concrete architecture. In [3,4] the authors propose a service development process whose activities originate from a service reference model, implemented as UML2 profile[3]. Following the same approach a middleware development process might originate from our SOM reference model.

In Section 5 we show how the specification of a lower-level (B3G) specific SOM model has been guided by our reference model (presented in Section 4) further accounting for specific motivation and goals of the PLASTIC project [16], as well as existing standards and specifications, i.e., WSA. In particular the audience of the model includes: (*i*) project managers that, while making decisions, want to reach agreements based on a consistent and common understanding; (*ii*) services and middleware developers for identifying specific features, specifying conforming models, and developing actual code; (*iii*) software analysts performing validation, verification, monitoring, etc; (*iv*) stakeholders who want to achieve a better understating of the concepts and benefits of using a SOM.

3 Towards a B3G SOM Reference Model

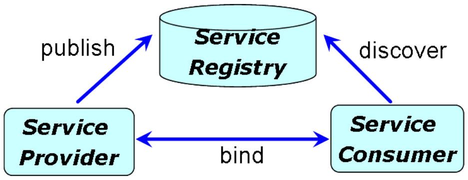

Fig. 2. Interaction pattern

---

[3] Available at http://gforge.inria.fr/projects/plastic-dvp/

In this section we introduce our model by highlighting the main concepts underlying SOA and the basic features that any SOM should provide to support the service-oriented interaction pattern (depicted in Figure 2). Such aspects are well known and deeply investigated in Literature [12,14,15,21], hence we briefly recall them and in the next section we fully present our model by concentrating on B3G related aspects. Our model will be specified by using a UML-like notation where rectangles represent concepts/entities (as used in SeCSE [12]) and relationships among them are used with the canonical meaning.

Within the service-oriented interaction pattern service providers publish their service descriptions into a service registry (i.e., a network addressable entity); service consumers query the service registry to discover services, hence obtaining the binding information (included in the service descriptions) that can be used for either static (development-time) or dynamic (runtime) binding. Generally, SOM should provide runtime support for service providers to deploy the service implementation into the execution environment and to publish service descriptions into the registry (see Figure 3). Services can be deployed and removed at any time, then the SOM should support runtime deployment and advertisement of arriving/departing services. On the other side, SOM should support service consumers to discover services by retrieving their descriptions and to bind/interact with the service providers for accessing the service implementations. Support should also be provided for combining autonomous networked services to realize more complex composite services. Furthermore, the "success" of service provision depends on the user perception of the delivered QoS, which (as a crosscutting concern) varies along several dimensions, including: type of service, type of user, type of access device, and type of execution network environment. The so called WS-* specifications cover (most of) these aspects and, in particular, the work in [13] is a promising attempt to establish a well-founded reference model for QoS.

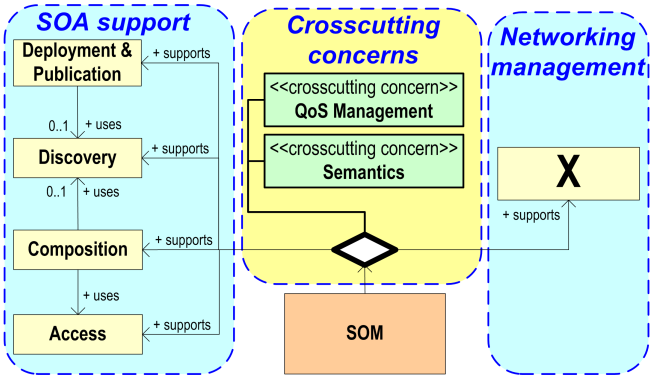

Fig. 3. SOM basic features

In addition to QoS, another fundamental crosscutting concern is the semantic aspect of service publication, discovery, interaction and composition [15]. For instance, to discover and select services a syntactic matching between the requested functionalities and the advertised ones is required but it is not enough. In fact, having semantic information within service descriptions would allow a machine to process and understand the “meaning” of the services description and hence support enanched discovery and automatic integration.

As already introduced, there is no attempt to abstractly model a SOM by comprehensively considering the complexity and the management of the next generation B3G networking environment (see Figure 3). That is, exploiting the SOA style in the context of B3G poses new requirements on the underlying SOM that has to also provide the proper abstractions concerning both (*i*) the network characteristics and the functionalities related to the management of the infrastructures, and (*ii*) the services' dynamic context of execution. In order to manage B3G networks, their diversity and richness must be made available and be exploitable at the service layer, where the service provision and consumption can be most suitably adapted. That is, B3G networks characteristics cannot be taken apart but they must be considered since the modeling phase [9]. To this extent, in the next section we raise up B3G concepts at the reference model level.

4 B3G SOM Reference Model

B3G networks pursues the convergence between wireless telecommunication networks and wireless IP networks by offering broad connectivity through various network technologies at once, thanks to the newest multi-radio devices like smart phones embedding, e.g., UMTS, WiFi and Bluetooth. Still, most applications running on those devices do not exploit the multi-radio networking capability in an integrated manner, and are commonly developed with a single network in mind. Leveraging the rich B3G networking environment poses new challenges, e.g., network selection, QoS negotiation, mobility management, context-awareness, primarily related to the network openness, and inherent heterogeneity and dynamism. A key challenge is then to manage in an automated and integrated way the available radio interfaces of the device. Such integrated management must ensure connectivity for the onboard applications and improve the exploitation of the networking environment. However, as already introduced, existent approaches to SOA and SOM modeling do not explicitly consider the networking environment as means for interaction between consumer and provider. In particular, the OASIS model generally specifies that the interaction is performed within the *execution context*, abstractly defined as all the business and technical “elements” that form a path between the consumer and the provider [14].

Explicitly considering B3G networks as execution context, the B3G-SOM should conveniently combine the heterogeneous networks in reach that, for various reasons (e.g., cost effectiveness, distinct administration and infrastructureless ad hoc interaction), are not integrated at the network layer (see Figure 4). In par-

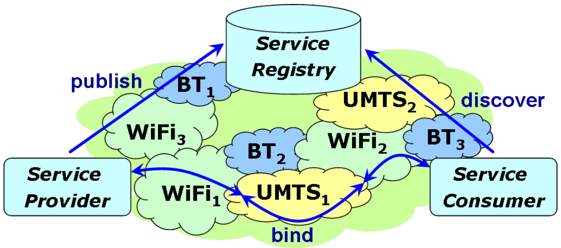

Fig. 4. B3G execution context

ticular, we introduce (see also Figure 5) the notion of `Network Infrastructural Path` between the service provider and consumer as combination of possibly different networks together with their network-level services. For instance, referring to Figure 4, a possible network infrastructural path is $WiFi_1 \Leftrightarrow UMTS_1 \Leftrightarrow WiFi_2$.

Hence, B3G-SOM should support services for being aware of the networking environment and for adapting to its changes. The B3G-SOM should be able to capture the various networks and observe their status, and abstract their properties in order to fully exploit the underlying network diversity without relying on any pre-established knowledge or infrastructure.

Valuable enabling technologies approaches to integrate different native network infrastructures are: (*i*) the Unlicensed Mobile Access [2] approach that allows real-time handover between access networks and (*ii*) the IP Multimedia Subsystem [1] approach that is a SIP-based solution for providing seamless roaming between WLAN and Cellular networks with a single user identity. These systems aim to integrate unlicensed wireless networks (e.g., WiFi) and 3G Cellular networks into a complex network, which enables subscribed B3G devices to seamlessly roam between the two; (*iii*) at lower level, the emerging IEEE 802.21 [8] is investigating on how to enable seamless handover among heterogeneous networks by defining a link-layer event-based signaling used for gathering information about the underlying networks. For the purposes of this work, these approaches have been investigated to understand how a B3G-SOM should comprehensively deal with the B3G execution context while exploiting the SOA style. In this direction, in the following we discuss how the supporting features provided by a not-B3G oriented SOM must be enhanced to cope with B3G.

Service Deployment & Publication – Leveraging B3G networks does not add any specific issue to the deployment activity, whereas it does heavily affect the publication one. In fact, SOA over B3G envisions networked mobile devices that, while moving along different networks, publish (some of) their deployed applications as services. The B3G-SOM shall then provide a proper publication mechanism that allows providers to publish their service descriptions with respect to the network currently accessed. That is, whatever discovery model (e.g., active or passive) and registry implementation are used, the key issue is to keep up-to-date the published binding information that, due to the (mobile) dynamic availability of services (arrival or departure), changes over time.

Service Discovery – While many Service Discovery Protocols (SDPs) have been proposed and widely adopted for various networking environments (Internet, home networks)[4], B3G environments add further issues arising from having multiple heterogeneous networks in reach. Multi-radio devices, empowered by B3G-SOM, should then support multi-networks communication - either at the network level (IP routing) or at the application level (network overlay) - in addition to an efficient and scalable dissemination and filtering of discovery requests/announcements. Moreover, (*i*) interaction problems might arise from the presence (at a given location) of different discovery protocols, data representation and communication formats; (*ii*) multiple instances of the same service may be available and hence an accurate service selection should be addressed by means of context and semantic information; (*iii*) a further stumbling block, strictly related to mobility issue, is that the discovery process cannot rely on devices permanently present (at given locations) acting as (central) service repository. That is, since devices cannot play the role of static service repository, the B3G-SOM should provide a mechanism for dealing with the availability (unavailability) of services due to devices joining (leaving) the network.

Service Access – In the B3G networking environment, handling of connectivity loss and eventual reconnection cannot be systematically assumed. Connectivity needs to account for the multiple links that may possibly exist between two nodes and that have different properties, in particular regarding their lifetime. In general service access in the B3G network requires dealing with the comprehensive management of mobility, combining solutions to: (*i*) exploiting the redundancy of (possibly) multiple network infrastructural path among consumer and provider; (*ii*) assessing the quality of individual network paths so as to allow selecting the actual path to be used for message exchange; (*iii*) handling vertical handover at the service level by locating new network locations of a given service instance, should its host no longer be accessible via the original links; (*iv*) handling service substitution by exploiting possible loose replication of services within the network, i.e., the existence of multiple networked service instances matching a given service description.

Service Composition – Autonomous networked services can be combined to realize a more complex, composite service. Service composition lies in introducing: (*i*) a language for specifying composite processes, (*ii*) methods and tools assisting the thorough design and validation of composite services, and (*iii*) middleware support for service composition, from the actual composition of services (out of networked instances according to a given composite process) to the execution of the composite service. Leveraging B3G does not directly affect service composition since it relying on service discovery and access discussed above.

Extending the work in [6], we model `B3G Network` as combination of telecommunication networks and IP based networks, either `Wired` or `Wireless` (see Figure 5). Wireless networks are categorized into three groups based on their

---

[4] Jini: http://www.jini.org/, UDDI: http://uddi.xml.org/

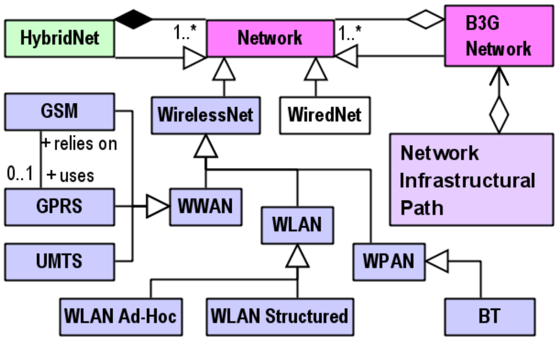

Fig. 5. B3G network model

coverage: (*i*) Wireless Wide Area Networks (`WWAN`) include wide coverage technologies such as 3G cellular (`UMTS`), Global System for Mobile Communications (`GSM`), General Packet Radio Service (`GPRS`), (*ii*) Wireless Local Area Networks (`WLAN`) include WiFi (`WLAN_Ad-Hoc` and `WLAN_Structured`), and (*iii*) Wireless Personal Area Networks (`WPAN`) such as Bluetooth (`BT`). We further identify `Hybrid Networks`, which generically indicate networks that integrate some of the above networks at the network-layer, as achieved by the network operators (e.g., integrated UMTS and WiFi connectivity).

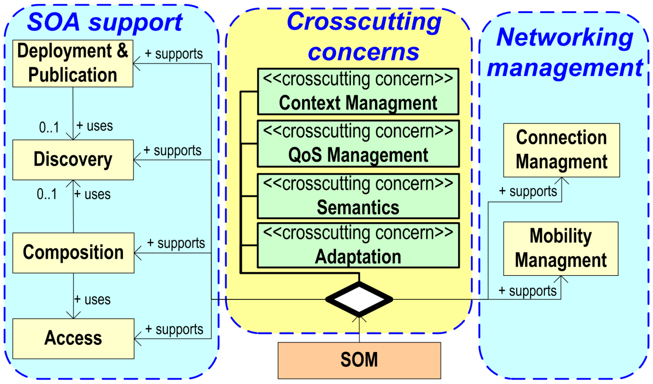

Fig. 6. B3G-SOM enhanced features

Figure 6 refines Figure 3 by also showing the specific features that should be supported by a B3G-SOM. In fact, exploiting the SOA style in the context of B3G requires the underlying SOM to also provide proper abstractions concerning both (*i*) the network characteristics and the functionalities related to the management of the multi-radio networking environment (*Connection Management*), and (*ii*) the services' dynamic context of execution (*Mobility Management* and

Context Management). Moreover, in order to manage B3G networks, their diversity and richness must be made available and be exploitable at the service layer, where the service provision and consumption can be most suitably adapted trough ad-hoc *Adaptation* mechanisms.

4.1 Connection Management

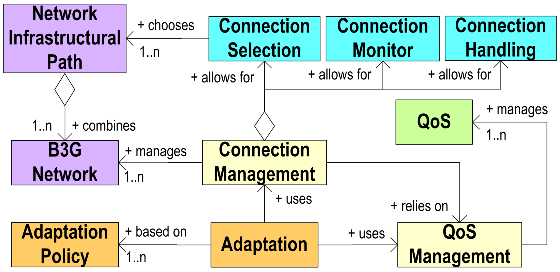

Fig. 7. Connection and Adaptation

From the discussion previously carried on, it is evident that taking benefit from the rich B3G networking environment is raising serious challenges related to the effective management of this complex environment. Indeed, while operating systems start embedding support for integrated management of multi-radio networks, the end-user is in charge of explicitly choosing and switching networks in most cases. This requires for a `Connection Management` (Figure 7) that permits to manage in an automated and integrated way the available multi-radio networks, composing the `B3G Network`, for ensuring connectivity and effectively improving the performance of the networking environment, both qualitatively and quantitatively. Due to rapid evolution of network technologies, connection managers should be developed in such a away that they can be easily and rapidly adapted to new incoming technologies. Still referring to Figure 7, the connection management should play the roles of: (*i*) `Connection Handling` for handling the connection requests coming from different applications (running on the same machine) and solving possible problems arising from conflicting network access requests; (*ii*) `Connection Selection` for choosing the best possible `Network Infrastructural Path` by fulfilling the minimal requirements of the connection request. Alternative paths should also be computed in case it is not possible to fully satisfy the request requirements or in case of connectivity loss; (*iii*) `Connection Monitor` for monitoring the network conditions and notifying applications about occurred changes.

4.2 Context Management

The common interest of most B3G applications in technologies that could achieve the above has led researchers to investigate multi-radio networking management below the application layer. The middleware presented in [17] provides a connection manager from which the application may get accurate knowledge about connectivity by monitoring the various links and related `QoS`, and then select the specific network connection(s) to establish. In this setting, network connections `Adaptation` should also be supported, possibly at different layers, according to some specific `Adaptation Policies` (Figure 7). For instance, in [18] adaptive network connections is realized at network layer, leading to fully transparent solutions for multi-radio networking on the end-user devices.

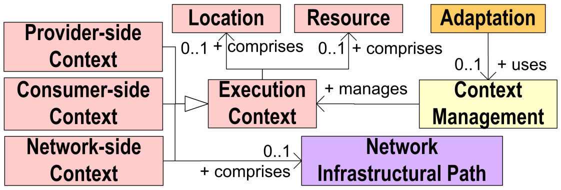

Fig. 8. Context and Adaptation

A further challenge in B3G networking concerns the `Context Management` (Figure 8). In fact, it is apparent that both consumer and provider service applications, running on multi-radio devices, should be context-aware and adapt to the variation of `Consumer-side Context` and `Provider-side Context`, respectively. However, considering only a (consumer and provider) device-based context management is not sufficient as mobile devices might be limited in the amount of sensed information - i.e., some kind of information is just available from the network. Therefore, context-aware adaptation becomes even more significant when considering the various network infrastructural paths, and thus the multiple `Network-side Contexts` of execution, which further keep changing over time due to user mobility. Thus, services should rely on sensed information from both the networks, and consumer and provider devices.

As already said, relevant characteristics of the networking environment and their variations should be monitored at runtime, so as to enable services to adapt accordingly. Such characteristics are not limited to specific networking parameters - e.g., bandwidth - but may concern the physical/logical `Location` and the `Resources` availability (Figure 8). Thus, the context manager should facilitate the management of context information being retrieved by different sources in a homogeneous way. For instance, location information can be supplied by GPS sensors or via information services the context-aware service has access to - e.g., triangulation data can be provided by a network operator. Thus, context management should allow for advanced operations on contextual data - e.g., comparing and reasoning on data for checking consistency and combining different types of data. Context management should also account for the dissemination of contextual data and for filtering mechanisms to protect personal privacy. Moreover, the knowledge of available resources and their measure - i.e., resource-awareness - should also be supported. The availability of certain types of resources is a context attribute and some specific mechanism is required to measure such an attribute (from a value source) for the purpose of declaring it in a context specification. This knowledge would allow for the adaptation of a service according to measured "resource bounds" (e.g., absence/presence, minimum/maximum value, range of values) so that service provision and consumption are free from runtime violations of such bounds.

4.3 QoS Management

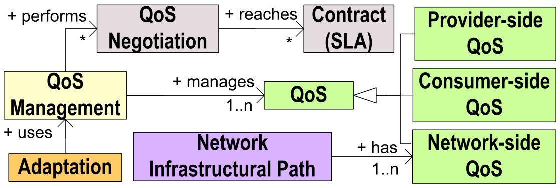

Fig. 9. QoS and Adaptation

Developers should be able to cope with this variable environment as much as they should be able to exploit it in order to realize services that deliver to the end-user the "best" `QoS` achievable, regarding both qualitative and quantitative attributes. To this end, services should be designed in a way that accounts for the QoS attributes of the overall execution context, where again the help of an underlying B3G-SOM becomes important for the `QoS Management` of `Consumer-side QoS`, `Provider-side QoS`, as well as `Network-side QoS` (see Figure 9). The notion of requested QoS (as part of the consumer-side QoS) and offered QoS (as part of both network- and provider-side QoS) should be used for establishing the Service Level Agreement (`SLA`) between the service consumer and the service provider. In fact, the `SLA` is an entity modeling the conditions on the QoS accepted by both the service consumer and the service provider [10]. `SLA` represents the establishment of a `Contract` that is influenced by the service request QoS and the execution context where the service has to be provided. When a new service request is formulated the QoS management has to possibly support `QoS Negotiation` to reach the agreement. The work in [5] proposes a modification to the service oriented interaction pattern in order to (possibly) reach the SLA at the end of the discovery phase. In [11] the authors present a framework for the development and deployment of (Java-based) adaptable service able to adapt to the current (hardware and software) resource conditions and provider-, network- and consumer-side QoSs.

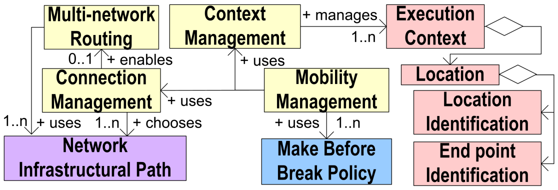

Fig. 10. Multi-network routing

4.4 Mobility Management

Yet another challenge for B3G SOA-oriented communication is `Mobility Management` in support of a seamless mobility (see Figure 10). However, seamless mobility in the B3G network is still at an early stage, independently of the layers in which it is tackled. The approaches in [1,2,8] strive towards the same goal: properly managing connections for seamless handover. In particular, there exist three types of handover: (*i*) horizontal handover allows consumers to seamlessly switch between different access-points of the same network (e.g., between two base stations of a WiFi network), (*ii*) vertical handover allows consumers to switch from a network of a given type to one of another type (e.g., from Bluethooth to WiFi), (*iii*) roaming allows consumers to roam outside of their home service area (i.e., to access a network managed by foreign network operators). While the horizontal handover is directly managed by the network infrastructure itself, enabling vertical handover and roaming, which involve a number of heterogeneous networks, poses a new set of requirements that must be faced by the B3G-SOM. The challenge is to effectively implement the so called `Make Before Break Policy` for preparing the vertical-handover (between two different networks) before the interaction between provider and consumer breaks. In fact, when switching from a given network to one of a different type, the device changes its location according to the new environment it is entering into. Changing the device's location affects also the status of all the devices that are currently interacting with it. Hence, when a host changes its point of attachment, the location must be updated (i.e., the internal status) in order to route packets to the new network. In this setting, location meaning is twofold: `End Point Identification` (e.g., the IP that address uniquely identifies a host in a given network) and `Location Identification` (e.g., the network in which the host is located). Then, since the location is the base of any connection, all the ongoing connections break (i.e., the handover affects the status of the interacting parties). Furthermore, as devices can bind various networks at the same time, two interacting parties might communicate through multiple network infrastructural paths and hence, an alternative path should be proactively selected by the connection manager before the interaction breaks. Note that, the support for `Multi-network Routing` plays a crucial role in choosing the best execution path serving a given interaction, as this significantly affects the quality of service at large (e.g., availability, performance with respect to both resource consumption and response time, reliability, security).

5 PLASTIC-specific Reference Model

This section presents a minimal excerpt of the the PLASTIC B3G-SOM that represents a specific B3G-SOM reference model (see Figure 1). As already said in Section 2, it has been derived for the PLASTIC project [16] and it's specification has been guided by the B3G-SOM reference model presented in Section 4. One of the main objectives for the PLASTIC middleware [7] is to develop a lightweight SOM for accessing and dynamically composing mobile, adaptable services in B3G networks. To this extent, the PLASTIC B3G-SOM builds upon the Web-based SOA so as to benefit from the pervasive nature of Web technologies that makes them available in most digital environments. Indeed, PLASTIC consider WSA as the primary technological building block for services. This is motivated by the fact that WSA is widely adopted and that it introduces key enabling technologies for service orientation mostly covering the overall development process.

The middleware supports *Multi-radio networking* (Section 5.1) and *Web Service-oriented Communication* (Section 5.2). *Multi-radio networking* enables the execution of services on the end-users' wireless handheld devices embedding various radio network interfaces. By exploiting the multi-radio networking, *Web Service-oriented Communication* basically enriches traditional functionalities of a SOAP engine. It allows for SOAP-based interaction over the B3G network, which includes both: (*i*) enabling access to services that may be in distinct networks thanks to multi-network routing and (*ii*) dealing with seamless mobility as long as the respective hosts of the given service's consumer and provider remain in reach via at least one network infrastructural path.

5.1 Multi-radio Networking

The `Multi-radio Networking` offers the abstraction of an integrated multi-radio interface by comprehensively composing the various networks in reach via the embedded radio interfaces, as identified by the `Multi-radio Device Management` functionality (see Figure 11). `Multi-radio Device Management` is in charge of managing `Radio Network Interface`s implementing relevant network-related functionalities, such as switching on/off the radio interface, connecting/disconnecting to/from the network, sending and receiving packets on the network. Moreover, `Multi-radio Device Management` is in charge of sensing the `Network-side Context` and retrieving `Network-side QoS`, which are further used by the PLASTIC B3G-SOM for routing purposes (see Section 5.2). `Network-side Context` identifies the characteristics of the networks in reach, such as number of active users, number of available services, security, transmission protocol, access policy, etc. This is of particular relevance, since the network context impacts upon the QoS represented by `Network-side QoS` (e.g., bitrate, signal strength, power consumption, network coverage).

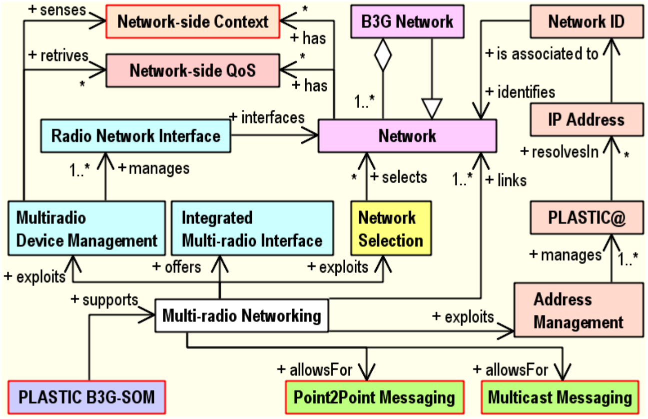

Fig. 11. Multi-radio networking

In this setting, it is crucial to consider network characteristics and decide which are the most appropriate networks regarding to the user needs and context. In other words, the multi-radio networking accounts for both the network characteristics (e.g., QoS, supported mobility and transport protocols) and the capabilities related to the access of the related infrastructure - if any - (e.g., security and authentication). Furthermore, as devices can bind various networks at the same time, service consumer and provider might communicate through multiple network infrastructural paths. Choosing the best path serving a given interaction is a key issue to deal with, as this significantly affects the QoS at large (e.g., availability, performance with respect to both resource consumption and response time, reliability, security). These are then used by `Network Selection` functionality for identifying and classifying the available networks and for choosing the most appropriate one(s) with respect to the requirements posed by the interactions that are carried on. However, this introduces new issues concerning to the network switching management and requires for a mobility support which deal with end point identification and location identification (see Section 4.4). Moreover, accessing multiple radio interfaces at the same time implies to access multiple (possibly overlapping) IP networks. This requires PLASTIC-enabled devices to bind different IP addresses, one for each active interface. To this extent, `Address Management` is in charge of uniquely identifying a given device through a B3G overlay address (simply referred to as `PLASTIC@` in Figure 11). The `PLASTIC@` of a given device is specifically the device's unique identifier, which resolves into the actual set of [`Network ID`, `IP Address`] bound to the device (at a given time). `Multi-radio Networking` allows for point-to-point and group-based message exchange between two interconnected multi-radio devices (identified by means of their `PLASTIC@`), i.e., `Point2Point Messaging` and `Multicast Messaging`. To this extent, `Multi-radio Networking` offers an abstract `Integrated Multi-radio Interface` by exploiting the underlying

`Multi-radio Device Management` and the `Network Selection` (refer to Figure 11).

In the following, we describe how the `Multi-radio Networking` can be exploited for achieving a `Web Service-oriented Communication` within the B3G networking environment.

5.2 Web Service-oriented Communication in B3G networks

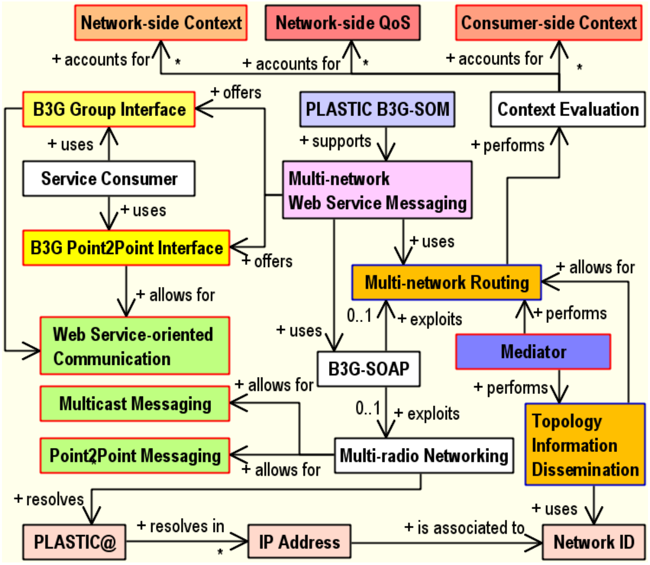

Fig. 12. B3G WS-oriented communication

The `WS-oriented Communication` of the PLASTIC B3G-SOM middleware is compatible with the WSA [19]. Since WS standards rely on SOAP messaging the middleware enables SOAP-based messaging and leverages the use of the `PLASTIC@` into the service description (i.e., endpoint identification). In order to fully exploit the capabilities of the B3G environment, PLASTIC B3G-SOM enables SOAP-based communication across multiple, loosely connected independent IP networks by relying on `PLASTIC@`s to identify Web services. Referring to Figure 12, the `Multi-network Web Service Messaging` functionality offers two interfaces for `Point2Point Interface` and `Group Interface` that together enable `Web Service-oriented Communication` in B3G environment. To achieve these two communication facilities, `Multi-network Web Service Messaging` uses `Multi-network Routing` and `B3G-SOAP`.

As detailed in Section 5.1, message exchange among devices may take place over the various networks linking the involved devices, and a device may itself be connected to, and reachable through, multiple independent networks.

However, in order to access services hosted in "distant" networks, the request of the client application has to be routed through one or more bridge nodes. Bridges are those (PLASTIC-enabled) devices which act as mediators. They are in charge of routing communication among independent networks composing B3G. To perform such task, bridge and network information must be exchanged so that an appropriate route can be selected. A `Mediator` supporting multi-network routing is thus started on each bridge node and performs `Topology Info Dissemination`. More precisely, the mediator component is in charge of routing `B3G-SOAP` messages between independent networks. This component is deployed on PLASTIC-enabled multi-radio devices and interacts with the PLASTIC `Multi-radio Networking` layer to get connectivity information, send and receive packets, and monitor `PLASTIC@`s addresses. The `Mediator` also supports context-aware multi-network routing by evaluating context information (`Context Evaluation`) when processing `B3G-SOAP` messages to decide whether or not to forward them. We specifically consider context information of the networking environment (`Network-side Context`) and client applications requirements (`Consumer-side Context`).

6 Conclusion and future work

In this paper we propose a reference model for B3G-oriented SOM (B3G-SOM) that brings together notions to be accounted while architecting middleware solutions for service-oriented applications over B3G. The model aims at establishing a common understanding of the support that should provided by a generic B3G-SOM and can be considered as a facilitator for designing architectures of specific B3G-SOM solutions. So far, there has been no attempt to tackle this issue in a systematic and reusable way, failing to offer to developers an "instrument" for identifying the relevant abstractions that need to be considered in order to develop B3G-SOM for services over B3G. To this extent, the proposed model brings together first-class concepts related to SOM with concepts related to the wide B3G and SOA domains.

As future work, following the approach in [4], we plan to implement a UML2 profile for our model by customizing an existing UML2 modeling tool. Our goal is to provide an integrated development environment that eases the design of conforming B3G-SOM models. In particular we will exploit a model driven development approach that, based on model-to-code transformation, shifts the focus of B3G middleware development from coding to modeling.

References

1. 3rd Generation Partnership Project (3GPP). IP Multimedia Subsystem (IMS). http://www.3gpp.org/.
2. 3rd Generation Partnership Project (3GPP). Unlicensed Mobile Access (UMA). http://www.umatoday.com/.

3. M. Autili, P. D. Benedetto, D. D. Ruscio, P. Inverardi, and M. Tivoli. A development process for context-aware adaptive services. In *Proc. of ARAMIS (ASE). To Appear*, L'Aquila, Italy, Sept 2008.
4. M. Autili, L. Berardinelli, V. Cortellessa, A. D. Marco, D. D. Ruscio, P. Inverardi, and M. Tivoli. A development process for self-adapting service oriented applications. In *Proceedings of the 5th International Conference on Service-Oriented Computing*, pages 442–448, Vienna, Austria, September 2007.
5. M. Autili, V. Cortellessa, P. D. Benedetto, and P. Inverardi. On the adptation of context-aware services. In *To appear: Service Oriented Computing: a look at the Inside (SOC@Inside'07), ICSOC*, Sept 2007.
6. M. Caporuscio and V. Issarny. An uml 2.0 profile for architecting b3g applications. In *3rd International Workshop on Rapid Integration of Software Engineering techniques*, Genevre, Swiss, September 2006.
7. M. Caporuscio, P.-G. Raverdy, H. Moungla, and V. Issarny. ubiSOAP: A service oriented middleware for seamless networking. In *Proc. of 6th ICSOC. To Appear*, 2008.
8. IEEE 802.21: Media Independent Handover Services. IEEE Standard under development. http://www.ieee802.org/21/.
9. V. Issarny, M. Caporuscio, and N. Georgantas. A perspective on the future of middleware-based software engineering. In L. Briand and A. Wolf, editors, *Future of Software Engineering 2007*. IEEE-CS Press, 2007.
10. J. Skene, D. Lamanna, W. Emmerich. Precise service level agreements. In *Proc. of the 26th ICSE.*, pages 179–188, Edinburgh, UK, May 2004.
11. M. Autili, P. Di Benedetto, P. Inverardi, D. A. Tamburri. Towards self-evolving context-aware services. In *Proc. of CAMPUS (DisCoTec)*, volume 11, 2008.
12. M. Colombo and E. Di Nitto and M. Di Penta and D. Distante and M Zuccal. Speaking a Common Language: A Conceptual Model for Describing Service-Oriented Systems, 2005.
13. OASIS. Web Services Quality Model (WSQM) TC. Committee Draft 2.0, http://www.oasis-open.org/, 2005.
14. OASIS. Reference Model for Service Oriented Architecture 1.0. Committee Specification 1, August, 2006.
15. M. Papazoglou, P. Traverso, S. Dustdar, and F. Leymann. Service-oriented computing: State of the art and research challenges. *Computer*, 40(11):38–45, Nov. 2007.
16. PLASTIC Project. PLASTIC Website: http://www.ist-plastic.org, 2005.
17. J.-Z. S. Riekki, J. Jurmu, and M.Sauvola. Adaptive connectivity management middleware for heterogeneous wireless networks. *Wireless Communications*, 12(6), 2005.
18. J. Sachs, L. Muoz, R. Aguero, J. Choque, G. Koudouri, R. Karimi, L. J. J. Gebert, F. Meago, and F. Berggren. Future wireless communication based on multi-radio access. In *WWRF*, 2004.
19. W3C Working Group. Web Services Architecture (WSA). http://www.w3.org/TR/ws-arch/, February 2004.
20. T. Zahariadis and B. Doshi. Applications and services for the B3G/4G era. *Wireless Communications, IEEE*, 11(5), Oct 2004.
21. U. Zdun. Pattern-based design of a service-oriented middleware for remote object federations. *ACM Trans. Interet Technol.*, 8(3):1–38, 2008.