



HAL
open science

Towards highly adaptable user-centric terminal architecture

Lucian Suciu, Jean-Marie Bonnin, Bruno Stévant, Karine Guillouard

► **To cite this version:**

Lucian Suciu, Jean-Marie Bonnin, Bruno Stévant, Karine Guillouard. Towards highly adaptable user-centric terminal architecture. WPMC 2004: 7th international symposium on wireless personal multimedia communications, Sep 2004, Abano Terme, Italie. <hal-02378892>

HAL Id: hal-02378892

<https://hal.science/hal-02378892v1>

Submitted on 25 Nov 2019

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.



HAL Authorization

Towards a Highly Adaptable User-Centric Terminal Architecture

Lucian Suci, Jean-Marie Bonnin, Bruno Stévant

École Nationale Supérieure des Télécommunications de Bretagne
BP78 2, rue de la Châtaigneraie, 35512 Cesson-Sévigné, France
e-mail: {lucian.suciu, jm.bonnin, bruno.stevant}
@enst-bretagne.fr

Karine Guillouard

France Télécom R&D, DMR/DDH Laboratory
BP59 4, rue de clos Courtel, 35512 Cesson-Sévigné, France
e-mail: karine.guillouard@francetelecom.com

Abstract—In the next-generation communication systems, the mobile terminals will be considerably more diverse than nowadays, and the users will have a greater choice of access technologies, offering different QoS, security, cost, and so on. However, the “best” decision to select an interface and an access network from many other possible combinations has to be taken. This decision will depend on information such as: the user preferences, the requirements from applications, the device capabilities, the performances and capabilities of the available networks, the network operators constraints, etc. This paper describes an add-on middleware which deals with interface automatic configuration and selection. Our goal is twofold: to provide users with “seamless” roaming between heterogeneous access networks, and to allow them to always stay connected through the “best” access network.

Keywords—heterogeneous access networks, profile management, selection decision algorithms, “always best connected”, adaptable and reconfigurable terminal architecture

I. INTRODUCTION

Some glimpses of future all-IP mobile architectures can be foreseen by carefully analysing the current trends. First of all, there is an increasing number of mobile computers and communicators with outstanding performances, such as smartphones, PDAs, tablet PCs, and laptops. Then, we notice a greater choice of access networks and simultaneous multiaccess of these networks, including WLAN (IEEE 802.11a/b/g), WPAN (Bluetooth, IEEE 802.15), MBWAN (i.e. IEEE 802.20), GPRS/UMTS, and so on. Finally, there is a need for much more advanced mobile services, comprising adaptive and self-configuring services, context awareness applications, user profiling and personalisation, etc.

This complex communication environment has already opened up new research areas which attempt to improve mobility (e.g. MobileIP and micro-mobility schemes), the AAA-Authentication, Authorization and Accounting (e.g. DIAMETER) and the QoS (see [1], [2]), ultimately the end-user experience. However, two basic requirements have clearly emerged. The first one mandates that users should be provided with seamless roaming amongst heterogeneous access networks (e.g. [3]), including simultaneous or successive connections to several access technologies. The second one states that the users should be allowed to always stay connected through the “best” access network ([4], [5]).

To satisfy these two goals we propose a user-centric terminal architecture which is adaptable and reconfigurable. Thus, regular users are able to create profiles that, according to the situation, allow them to access the most suitable network. The terminal is adaptable because it always takes into account the current context (i.e. users preferences, terminal resources, networks conditions and applications needs) and it tries to adjust to the context without the user’s intervention. It is reconfigurable because either the user and the administrator can redefine their preferences, add new configurations for network interface cards or subscribe to new services; then, the terminal will put into practice these changes.

However, we do not directly address the design of adaptive applications (e.g., see [6]), but we provide a clearly defined

API towards them. We also try to set forth a solution which works properly for applications which are unaware of our add-on middleware.

The rest of the paper is organised as follows. We first present the related work in Section 2. Then we describe our proposed architecture in Section 3. In the next two sections we give more details about the Profile Manager (PM) and the Selection Decision Algorithm (SDA). Section 6 presents the implementation and the results obtained so far. We reveal the future work and conclude with Section 7.

II. RELATED WORK

There is a growing number of activities related to the definition and the handling of profiles. In the following paragraphs we present some of the most relevant of these research and standardisation efforts.

In [7] and references therein, the notion of Generic User Profile (GUP) is treated. The GUP is the collection of data which is stored and managed by different entities such as the user equipment, the home or visited networks, and which affects the way in which the user experiences the different services offered. The WAP User Agent Profile (UAProf) (see [8]) is concerned with capturing classes of device capabilities and preference information. These classes include general software and hardware characteristics of the device, as well as information about the network to which the terminal is connected. The Composite Capability/Preference Profiles (CC/PP) framework [9] is yet another mechanism for describing the capabilities and preferences associated with users and user agents accessing the World Wide Web.

The Information Society Technologies (IST) AQUILA project tries to provide dynamic control to DiffServ based traffic. One part of this project is to define the application profiles which contain the concrete application descriptions. Moreover, the IST-TRUST project tries to understand the users’ requirements for the Reconfigurable Radio Systems. It also defines a layered architecture that contains a policies and profiles management component. These profiles are further refined within the IST-SCOUT project.

On the other hand, in the recent years, the interface selection problem for a multi-interface terminal, communicating in a heterogeneous wireless environment has gained in importance ([10], [11], and [12]). Furthermore, the [13] proposes a Mobile Policy Table, which supports packet delivery over multiple networks simultaneously, and it adaptively selects the most appropriate network according to the characteristics of each traffic flow. In [14] professor Katz et al. try to define the “best” access network and “when” to handover to it. They do not support per-flow decision and their mechanism does not include a simultaneous use of multiple networks. The Parametric Cell Switching, as described in [15], uses different criteria in the handoff decision, i.e., the signal-to-noise ratio, the round trip time, and the price of using an access

network. However, it does not take into consideration the user's policies and the applications' needs.

The IST-CREDO project proposes an architecture which provides optimal IP-based services through a composite radio environment (including GPRS, WLAN and DVB segments), on the basis of quality and cost criteria. A key component of the terminal, named Terminal Station Management System (TSMS), undertakes the task of selecting the appropriate radio technology to be used for a service session by taking into account service and user profiles, as well as QoS and cost criteria.

The work for 3GPP-WLAN interworking is also ongoing in standardisation bodies, including access network selection (see [16]). In the case of 3GPP to IEEE 802.11 WLAN handover, the terminal stores a list of preferred SSIDs provided by the home network operator and it also maintains a list of the user's preferred SSIDs. The terminal will try first to associate with an available access point from the user SSID list (the priority order is defined) and, if it does not succeed, it will try the SSIDs from the operator list. In the WLAN to 3GPP handover case, the WLAN node should always indicate its home network (i.e., HPLMN) in its Network Address Identifier in the EAP-Response/Identity message. If the connection with the home network is not possible and the terminal is on the automatic selection mode, then it employs the user's list of PLMNs (some priority order should be set) or it may use the operator's list of PLMNs. There is also a manual mode, i.e., the terminal indicates to the user a list of available SSIDs or available PLMNs and the user makes a manual selection. However, the standard does not specify how to set the priorities within these lists (yet, it says, e.g., that the signal quality should not be used as a parameter in the case of PLMN selection).

Then, in the Candidate Access Router Discovery (CARD) draft proposed by IETF Seamoby Working Group (see [17]), the existence of a Target Access Router (TAR) selection algorithm is recognised. However, its specification is out of the scope of the aforementioned draft.

To conclude this section it can be said that there are many research efforts to define and handle useful profiles, and to invent multi-criteria selection decision algorithms. However, less work has been done to integrate the profile management with the optimal interface selection problem.

III. UBIQUE FRAMEWORK

Nowadays, it is usually the user who supervises a communication after initialisation and, if several access possibilities exist, the user decides which one to employ. Our approach is to investigate the network interfaces' capabilities, the access networks and the applications' limits, and also to interact with the users for obtaining their preferences.

At the end, we must be able to extract in a structured way all these capabilities, requirements, and preferences in profiles. The decision on whether to use or not an interface will be based on these profiles. Our aim is to provide, in the "best" possible way, simultaneous or successive connections to several access technologies within multi-interface terminals.

Fig. 1 shows the envisaged terminal architecture, its internal components and the possible interactions amongst them. Layers from above the Ubique middleware gather user/administrator preferences, handle applications requirements, and detect current terminal capabilities.

Layers from below the middleware detect the available networks, provide real-time information about the interfaces and the access networks capabilities, and make on-demand network interface configuration.

Furthermore, a separate "low-level" component handles the selection execution process, i.e., it actually maps the application

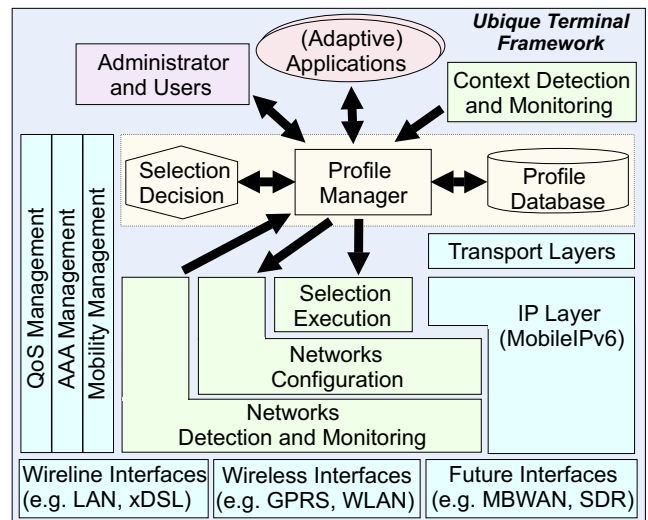


Fig. 1. Ubique architecture

flows on specific interfaces. The Ubique middleware will control all these "low-level" layers by initiating network configurations and performing interface selection decisions.

It can be seen from Fig. 1 that the Ubique middleware and "low-level" layers also need to interact with the transversal functions, such as mobility, QoS and AAA.

Nevertheless, we chose to separate the architecture in different functional blocks because it facilitates the implementation and testing, and it also permits the integration of new network configuration components (e.g. Context Transfer Protocol), of better selection decision algorithms, of novel network monitoring techniques, or of fine-grained selection execution (e.g. Per-Flow Movement).

Furthermore, we have preferred this modular approach because the selection decision algorithms do not need to know how parameters from profiles are collected or how selection decisions are enforced. Yet, because selection decisions heavily rely upon the types of parameters and their values, we prefer to keep a tight coupling relation between the SDA and PM.

We used clearly defined interfaces between the Ubique middleware (more specifically the Profile Manager) and the external blocks. For example, as the Network Detection and Monitoring component needs to deal with various access technologies, it has to convert the information into a generic format which is then sent to the PM. In fact, it is this generic format that allows the comparison of the capabilities of various access networks.

Then, the Profile Manager needs to periodically inform the Selection Execution module about the preferred network interfaces to be used for communication. As for the "high-level" entities, the PM provides bi-directional interfaces towards them, i.e., users and applications have to make their requirements and have to be informed about the changes within the system.

We conclude this section with an important observation, not shown in the Fig. 1: the applications' data traffic is not passing at all through our middleware, i.e., the applications will send and receive their data traffic directly to, respectively from, transport layers as usually, through the standard socket API.

IV. PROFILE MANAGER

As it can be seen from Fig. 1, an important part of the proposed terminal architecture will be dedicated to the definition and the management of profiles. Profiles are files stored in a Profile Database (PDB) which summarise key information about the components of the system and its interactions with the environment (i.e., users, applications, access networks, or service providers).

More specifically, the profile mechanism serves the following purposes:

- to automatize the selection of an interface and access network by maintaining all the necessary information in the PDB;
- to assist the Selection Decision Algorithm when it makes the choice of the “best” access option by taking into consideration the applications’ requirements and users’ preferences;
- to inform the adaptive applications about the unsteady capabilities of the interfaces and access networks;
- to set forth a solution which works properly for the applications unaware of modifications done to underlying layers.

We propose two kinds of profiles within the Profile Database: generic profiles and specific profiles. The generic profiles describe what information could be stored in the various types of profiles (i.e., they can be seen as patterns or schemas). Then, the administrators (or, let us say, the network operators), the users, and the applications can instantiate a generic profile to build specific profiles corresponding to specific cases.

Four generic profile types exist in the Profile Database:

- 1) Preferences and Resources Profile (PRP): it specifies how the system should behave relying on the available resources or on the current context. The information contained in this profile is provided by many sources: the administrator of the system, the users, and the applications.
- 2) Flow Description Profile (FDP): it mainly contains the QoS-related parameters. One part of this profile comprises the application QoS requirements and the other part contains the values of the QoS parameters monitored by the system.
- 3) Access Network Profile (ANP): it specifies all information (both Layer 2 and Layer 3) needed to access the network. In addition, it may contain information related to cost and security.
- 4) Network Interface Profile (NIP): it comprises the network interface card technical specifications, as well as the statistics obtained during runtime.

The information from different profiles is managed in a uniform and extensible manner (using the XML paradigm for example). This will ease the refinement of profile structure, as users’ preferences, applications’ requirements or network technologies will evolve.

It can be said that the Profile Manager acts as a dispatcher and a coordinator within our architecture. As depicted in Fig. 1, the PM interacts with all entities involved in profiles supplying and it knows which information has to be stored in the Profile Database. Furthermore, after updating the PDB, the PM determines if the Selection Decision Algorithm needs to be informed or not.

V. SELECTION DECISION ALGORITHM

Most of the selection decision algorithms (also known as handover algorithms) take into consideration just one selection criterion, usually Received Signal Strength (RSS) or Signal-to-Noise Ratio (SNR). More recently the multiple selection criteria algorithms have emerged as a better alternative and we have also taken this research path.

As it has already been described, the Profile Manager determines if the Selection Decision Algorithm (SDA) needs to be informed or not about the changes within the terminal. The PM must filter the plethora of various triggers; otherwise, the SDA could be activated very often and it will exhaust the CPU or the terminal battery.

Based on the triggers received from the PM, the Selection Decision Algorithm interrogates the Profile Database and starts its computational procedure. Thus, various profiles are used as input by the SDA in order to select the “best” interface for an application flow.

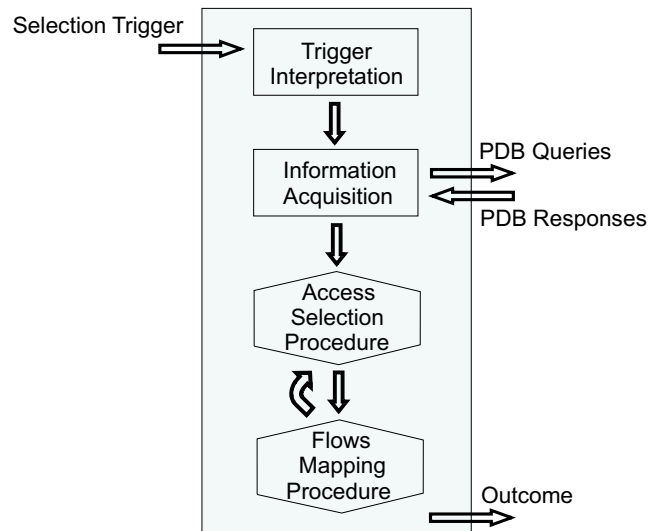


Fig. 2. Selection Decision Algorithm

The current SDA procedure is to define network interface cost functions (i.e., for the access selection), which are minimised, and application utility functions (i.e., for mapping the application flows), which should be maximized. Then, to solve this multiple-goal problem, we use the classical weighting objectives method. The general Selection Decision Algorithm mechanism is depicted in Fig. 2.

The suggestion whether to use or not an interface and an access network can be done on a flow-per-flow basis, or the SDA can propose only one interface for all flows. Furthermore, in order to cope with the various constraints and satisfy the users and applications requirements, the Selection Decision Algorithm could decide that it is better to (re)map some of the existing flows on other interfaces.

In the future, the SDA could also request network interface activation if needed (including probing, association, and authentication), or it could suggest possible adaptations in various layers and this may include QoS negotiations, header compressions, software defined radio configuration, and so on. As it is envisaged today, the algorithm provides flow mappings only for the outgoing flows and makes no assumption about the incoming flows. Yet, a distributed SDA could also suggest, together with the correspondent nodes and/or the networks, a global flow mapping which takes into consideration the incoming flows as well.

We can say that in our approach the selection of an interface and an access network is done not only to preserve a connection, but also to optimise the resources and to satisfy the user and the application’s expectations. Specifically, the Selection Decision Algorithm outcome consists of two lists of interfaces for each application flow: an ordered list of preferred interfaces and a list of forbidden interfaces.

It can be pointed out that the current SDA only offers middle-term (i.e., from hundreds of milliseconds to few seconds) adaptations. Nevertheless, our framework supports an extensible interface with the Selection Execution component, which actually maps the flow and makes short-term (tens of milliseconds) adaptations when needed. This will happen, for example, when the RSS/SNR drops below the communication sustainable limit and the Selection Execution module immediately re-maps the flows on the next preferred interface from the list provided by the SDA.

VI. IMPLEMENTATION AND FIRST RESULTS

To implement and test the proposed architecture we have chosen two end-user terminals: IPaq 3970 running Familiar Linux, and

Dell Latitude C610 running Debian Linux. To support the L3 mobility we have installed and configured the Mobile IPv6 for Linux distribution (i.e., MIPL). As wireless access technologies, we have 802.11b and Bluetooth access points and a commercial GPRS network. As context detection, we monitor the battery status for the time being.

All four types of profiles were implemented in XML and XML Schemas. Profile handling within the Profile Manager and the Profile Database, and the Selection Decision Algorithm were implemented using Ruby1.8, which is a portable, lightweight, object-oriented scripting language. The PM, PDB and SDA inter-modules communication is done through the DRB (Distributed Ruby), and with other modules through an XML-RPC-like protocol. To detect and configure the network interfaces we use bash scripts; thus, in the current implementation we stay out of the kernel space.

Our current choices for the selection initiation triggers are:

- preferences and resources changed: it is generated when the PRP has been modified by the user or the context has changed (in our case the battery status);
- flow created or flow deleted: a new application flow is started or terminated;
- flow parameters changed: generated when per-flow monitored parameters have changed their values;
- interface ready or interface not ready: that is, when a network interface successfully associates with the access network and it is ready for communication, respectively, when an interface lost its connection with the access network;
- interface parameters changed: it is generated when per-interface monitored parameters have changed their values.

As decision criteria, we use from the active PRPs (i.e., the current user and the running applications) the list of forbidden access networks for user and applications, the monetary cost versus QoS goal parameter, and the required security level for each application. Then, from ANPs we take the cost per byte and the security level offered by a particular access network. Finally, from NIPs we obtain the average bit rate, the monitored bit error rate, and the measured delay per network interface, and from FDPs we use the minimum necessary bit rate, the supported bit error rate and the maximum delay for each application flow.

The results obtained so far are encouraging. The middle-term selection latency (i.e., starting from selection initiation trigger till the phase when the decision to (re)map a flow was taken) depends on the trigger type and it takes 2-3 seconds on the PDA and under a second on the laptop. In future, we intend to offer a complete evaluation of our architecture by taking into account:

- the end-user satisfaction (this can be somehow assessed by the number of covered use-cases);
- the selection cost gain (in general the monetary cost gain should be evaluated, but the aggregate bandwidth or the security gain should not be overlooked);
- the number of flow re-mapped in similar conditions and during a well-defined period of time;
- the number of cut off flows.

Furthermore, as we already have a full functioning prototype, it is relatively easy to update the profiles with new parameters and to test new selection algorithms.

VII. CONCLUSIONS AND FUTURE WORK

In this paper we have presented an adaptable and reconfigurable architecture for future end-user terminals supporting multiple access network interfaces. Our first goal was to provide seamless access to heterogeneous networks, including simultaneous or successive connections to several access technologies. The second aim was to allow the user to always stay connected through the “best” access network. The proposed architecture includes adaptation

mechanisms and relies on tight interactions amongst the different layers, from the application layer to the data link layer.

We argue that the handover mechanisms within the future heterogeneous wireless networks will be much more complex than nowadays, i.e., they need to be divided into two parts: a contributory middle-term handover algorithm (i.e., the one which takes into account various profiles, as presented in this paper) and an essential short-term handover algorithm (i.e., the one traditionally based on the RSS or the SNR).

Our on going work focuses on profile management and selection decision algorithms. The various profile types need further refinements, for example uniform monetary cost representation within access network profiles. In addition, more selection strategies need to be investigated and the most promising will be implemented and evaluated.

Specifically, three areas should be thoroughly examined: the initiation triggers (i.e., when to start the selection algorithm), the decision criteria (i.e., which parameters to use), and the selection algorithms (i.e., how to usefully combine the chosen parameters). In addition, more tests are needed in order to grasp all benefits of the proposed terminal architecture. Finally, the impact of the proposed framework on the transversal functions (i.e., mobility, QoS, and AAA) needs to be further investigated.

REFERENCES

- [1] X. Fu, H. Karl, and C. Kappler, “QoS-Conditionalized Handoff for Mobile IPv6”, In E. Gregori et al. (Eds.): *Networking 2002, LNCS 2345*, May 2002
- [2] D. Wisely, et al., “Transparent IP Radio Access for Next-Generation Mobile Networks”, *IEEE Wireless Communications*, August 2003
- [3] T. Zahariadis, “Trends in the Path to 4G”, *Communications Engineer*, February 2003
- [4] E. Gustafsson, A. Jonsson, “Always Best Connected”, *IEEE Wireless Communications*, February 2003
- [5] M. O’Droma et al., “‘Always Best Connected’ Enabled 4G Wireless World”, *IST Mobile and Wireless Communications Summit 2003*, June 2003
- [6] F. André and B. Deniaud, “Multimedia Systems Adaptation in Mobile Environments”, *IASTED International Multi-Conference. Applied Informatics (AI 2003), Parallel and Distributed Computing and Networks (PDCN’2003)*, February 2003
- [7] 3GPP TS 24.241 V0.4.0, *3GPP Generic User Profile (GUP) Common Objects; Stage 3*, August 2003
- [8] WAP Forum WAP-248-UAPROF V20, *User Agent Profiling Specification*, October 2001
- [9] W3C Working Drafts, *Composite Capabilities/Preference Profiles (CC/PP): Structure and Vocabularies*, March 2003
- [10] K. Pahlavan et al., “Handoff in Hybrid Mobile Data Networks”, *IEEE Personal Communications*, April 2000
- [11] T. Park, A. Dadej, “Adaptive Handover Control in IP-based Mobility Networks”, *1st Workshop on the Internet, Telecommunications and Signal Processing (WITSP’02)*, December 2002
- [12] W. Zhang, J. Jaehnert, K. Dolzer, “Design and Evaluation of a Handover Decision Strategy for 4th Generation Mobile Networks”, *57th Semiannual Vehicular Technology Conference VTC 2003-Spring*, April 2003
- [13] X. Zhao, C. Castelluccia, and M. Baker, “Flexible Network Support for Mobility”, *4th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 1998)*, October 1998
- [14] H. J. Wang, R. H. Katz, and J. Giese, “Policy-Enabled Handoffs Across Heterogeneous Wireless Networks”, *2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA ’99)*, February 1999
- [15] T. W. Andersen, A. Lildballe, *Seamless Handoff in Mobile IPv6*, Master Thesis, Aalborg University, June 2001
- [16] 3GPP TS 24.234 V1.4.1, *3GPP System to WLAN Interworking; UE to Network Protocols; Stage 3*, May 2004
- [17] M. Liebsch et al., *Candidate Access Router Discovery*, Work in Progress, Internet Engineering Task Force, draft-ietf-seamoby-card-protocol-06.txt, December 2003