The Simplicity project: improving ease of use and personalization of ICT services

Nicola Blefari Melazzi
DIE, Università di Roma "Tor Vergata", Via del Politecnico, 1, Roma, 00133, Italia
blefari@uniroma2.it

Abstract

As technology develops, people are using an ever broader and heterogeneous range of ICT devices and network-based services. The result is an enormous burden of complexity on the shoulders of users, service providers and network operators. Excessive complexity, in turn, creates obstacles to effective exploitation and acceptance of beyond 3G systems and paradigms such as ambient intelligence, context-aware services, pervasive computing and novel access technologies. The goal of the Simplicity project, supported by the European Union, is to reduce this complexity by: i) providing automatic customization of user access to services and the network; ii) automatically adapting services to terminal characteristics and user preferences; iii) orchestrating network capabilities.

This short paper presents the Simplicity project, its initial outcomes (including a sketch of the system architecture), and tries to identify future research directions in this challenging field.

1. Introduction

The Simplicity project is a European Union program, scheduled to run for two years, from January 2004 to the end of 2005 [1]. Simplicity stands for Secure, Internet-able, Mobile Platforms LeadIng CItizens Towards simplicity. The project acronym intends to convey the very aim of the project: develop and evaluate a series of tools, techniques and architectures enabling users to customize and use devices and services with minimal effort.

A trans-European project, Simplicity brings together a combination of expertise from 11 major European industrial organizations, network operators, SMEs, research labs and universities [1].

This paper describes the challenges and motivations that led to the proposal of the Simplicity project and sketches the preliminary ideas developed in the first months of the project lifetime. As a consequence, this work should not be regarded as a technical paper, providing specific results. Rather, the aim of the author is to present an application scenario, some hints for future research directions and possibly to stimulate discussions in this challenging field.

The Simplicity project started from a vision: users today employ a variety of different terminals and devices to access a range of different "services" in the home, in buildings or in public spaces, for example, communications, computing capabilities, security support etc. Some services may be as simple as remote control of an entertainment device (e.g., a television) via a wireless link, or access control to a building. Others can be very complex and may require location awareness, QoS support, message exchanges with network databases, structured interaction with remote networking devices (e.g., media gateways), etc. The emergence of new research areas, such as pervasive computing, will further increase the diversity of the devices and services with which users have to deal.

But already today, users who attempt to exploit the services on offer have to deal with multiple procedures for configuring devices, multiple authentication mechanisms and passwords, multiple billing and payment procedures, multiple access technologies and protocols. This creates an enormous burden of complexity (as well as the physical burden of carrying different devices). This complexity is likely to limit the effective exploitation of the wide range of access, virtual reality, ambient intelligence and context-aware solutions currently under study and development. It will deepen the digital divide, making it difficult for non-technical users to benefit from new developments. And of course it will also create difficulties for network operators, who will be forced to manage the complexity of a multi-access networking environment.

The strategic goal of Simplicity is to simplify the process of using current and future "services". More specifically, the project aims to design and deploy a brokerage level allowing i) easy personalization of

services to match user preferences and needs, ii) seamless portability of services, applications and sessions across heterogeneous terminals and devices, and iii) smooth adaptation of services to available networking and service support technologies and capabilities.

With these goals in mind, Simplicity will:

- Describe user scenarios and business models for the Simplicity approach;
- Explore new brokerage mechanisms and policies in a multi-access networking environment;
- Design a universal multi-application Simplicity Device to provide users with a simple and uniform mechanism for customizing services and terminals;
- Validate feasibility and benefits of the Simplicity approach within a test-bed.

As regards the organization of this paper, we first introduce the driving concepts of the project; we continue by presenting near term and medium term scenarios for the deployment of our solution; we sketch the overall system architecture and we conclude by discussing future research directions.

The reference list is short and contains only project papers. This is because the project produced a through analysis of the state of the art of related technologies, standards and works, which is available in [2]. The document [2] quotes more than 120 references. An extract of this deliverable is published in [3]. Finally, more details on architectural and functional issues, and related references, can be found in [4], and, summarized, in [5]. Some selected user scenarios are described in [6].

2. Driving concepts

An important feature of 2G wireless systems, e.g., GSM, is the portability of user identities between different terminals (i.e., mobile phones). In what follows, we propose a generalization of this concept, allowing users to move seamlessly between different distributed applications and services, using heterogeneous networking technologies and devices. This approach, we suggest, could provide a user-friendly solution to the challenges posed by a diverse service and technology environment.

The personalization concept is based on a "user profile". In our view, each user will be provided with a personalized profile, providing access to different services, perhaps using different classes of terminals. Creating and maintaining the user profile will involve also the automatic processing of behavioural information (though the user will be able to switch off automatic storage and/or delete specific information). More refined policies on how to handle specific types of personal information will be part of the user profile and will be controlled by the user. Full control of personal data, security of information, and user privacy are key issues for the Simplicity approach.

The personalized user profile will allow: i) automatic, transparent customization and configuration of terminals/devices and services; ii) uniform mechanisms for recognizing, authenticating, locating and charging the user; iii) policy-controlled selection of network interfaces and applications services. Thanks to the profile, users will also enjoy the automatic selection of services appropriate to specific locations (e.g., the home, buildings, public spaces), the automatic adaptation of information to specific terminal devices and user preferences, and the easy exploitation of different telecommunications paradigms and services.

Depending on users' characteristics, preferences and abilities, personal profiles could take the form of i) a standard profile defined by a Service Provider; ii) a pre-defined template whose parameters can be configured by the user; iii) an open profile designed by the user by using a high-level description language.

The user profile will be stored in a so-called Simplicity Device (SD). Alternatively, storage on the Simplicity Device might be limited to a "pointer", making it possible to download the whole profile from the network. Though it seems natural (from our own everyday experience of 2G systems) to think of the SD as a physical device (e.g., an enhanced SIM card, a Java card, a Java ring, a USB stick, a sensor, etc.) the SD could also be implemented as a network location or a software agent. If the SD is a physical device, users could personalize terminals and services by the simple act of plugging the SD into the chosen terminal.

One of the main novelties of the SD is that it is not tied to a single networking environment, or to a single class of user terminals. The SD will provide all the information necessary to adapt services to the characteristics of the terminal, the nature of the environment and the user's personal preferences.

This means, for instance, that:

- different users plugging their SDs into the same laptop will see different working environments, file systems, software tools, connection services, etc;
- a given user who plugs the SD into different terminals will always see the same personalized working environment (adapted to the characteristics of the terminal);
- users will be able to suspend and resume running applications/sessions by simply unplugging and plugging the SD; when the user moves from one terminal to another the application/session automatically adapts to the new context.

3. Simplicity Device: near term and medium term scenarios

The key role of the Simplicity Device is to store user profiles, preferences and policies (and possibly user data), in a secure way. Ideally, the SD should meet the characteristics listed in Table 1 below.

Table 1. Ideal characteristics of the SD

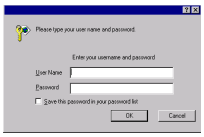
Characteristic	Comment
A "universal" device, compatible with a broad range of terminals	This is difficult, since we are talking of terminal devices ranging from phones to PCs to home equipments
Small, light-weight and low cost	Not exceeding a credit card
High level of user acceptance	Users must think of it as a familiar and trusted object
Include processing power	Needed to i) run security algorithms; ii) start user sessions when physically plugged into terminals, without necessarily requiring actions on the terminal itself (bootstrap function); iii) complement terminal host capabilities, if needed (e.g., a display or an actuator)
Largest possible storage capability	To store user data as well as profiles and preferences
Easily configurable	To allow users to quickly and simply install and configure profiles, perhaps via a GUI
Able to become "virtual"	To give users the choice of having a physical or a virtual personification of their preferences (e.g., network location or a software agent)

At first glance, Table 1 may be seen as nothing more than a wish list. It is nonetheless perfectly possible with current technology. The real issues are the following: i) will manufacturers be willing to combine all the facets of our ideal Simplicity Device into a single object? ii) will all terminal devices have slots to accept this SD? iii) will this SD enjoy widespread user acceptance?

On the one hand, we are confident that current trends go in the direction traced by this approach. On the other, we would like to find an easily implemented solution, which we can bring to market as quickly as possible. Thus, as a project, we want to follow two parallel routes: a mid/long term one and a short term one. In turn, the mid/long term solution includes two sub-options: i) assume to have a single object with all the aforementioned characteristics; ii) design a Simplicity Device concept able to be "adapted" to several instances of an idealized single solution.

To specify the mid/long term scenario and to identify a short term solution, it is useful to analyze devices that are currently available "off the shelf", to play the role of Simplicity Device. In this light, Table 2 summarizes pros and cons of some of these devices.

Table 2. Off the shelf "Simplicity" devices

Candidate device	Pros&cons
Smart Card 	☺: it has processing power ☹: limited storage; difficult to plug in into current devices; hard to say if future PCs, home equipments and phones will have a reader
USB stick 	☺: huge amount of storage; easy to plug in into PCs; ☹: no processing capability; some PDAs use USB, phones are more difficult (some phones have a USB "slave" port)
Secure Digital memory card 	☺: similar to a USB stick, but perhaps easier to use in PDAs and mobile phones; ☹: no processing capability; difficult for phones.
Virtual card (or biometric identification)   	☺: very "light-weight" ☹: no storage nor processing capability in itself; requires Internet connection; may require "smart" users; difficult to envisage the interaction with mobile phones and other relatively "dumb" devices; security issues: will users consider it a trustable object? In case of biometric identification, similar considerations apply, added to the necessity of adaptors.
GSM SIM 	☺: users think of it as a familiar and trusted object; ☹: not easy to plug in into current PC and laptops; if the same SIM is used for the

	mobile phones and for other devices, it would be necessary to plug the SIM in and out the phone; this could be annoying for users; limited storage; this approach would require cooperation from the mobile network operators with implication on the SIM ownership model, the overall business model and, last but not least, the issue of (open) interfaces between SIM and mobile phones.
Sensor (e.g., Mote with TinyOS) 	☺: easy and powerful interaction with the environment ☹: necessity of adaptors.

Based on Table 2, we can propose an ideal categorization of our SD in the following classes:

- Storage and processing devices: include a full powered CPU and related run-time memory, a crypto co-processor or software encryption facility, storage capacity for information (profiles, preferences, policies, etc.) that the user wishes to save on the SD, a communication interface;
- Pure storage devices: as above without CPU and built-in encryption capability;
- Pointer devices: would be a combined solution coupling a hardware SD with built-in encryption capability, without CPU. They would contain just enough memory to point to another location containing all other Simplicity-related data;
- Virtual devices: requiring network devices and applications that store SD data in a secure way and a login interface; the latter could be a plain old user+password one or could be based on biometric or sensor technologies.

The mid/long term choice for the SD could be a storage and processing device, eventually coupled with a virtual alternative, or a single abstract solution with different implementation alternatives, according to user preferences.

As regards the short term scenario, to find a solution, we have to take the concept of the Simplicity Device to extremes. The original idea of the SD was to store user preferences and data in a single device (possibly smaller than a laptop…), allowing users to personalize several different kinds of terminals/devices as well. This would take us towards a storage and processing device. We could go further along this path and arrive at a virtual card solution, but in this way we would loose the concept of the SD as a physical device: a distinctive pro in many situations, and for many (non-technical) users. What can we find in between a completely immaterial solution and a powerful-yet-small-card with storage and processing power? What we are looking for, at least in a short term perspective, is a "trigger": a device that can activate, prompt, and elicit reactions from more powerful components with more memory. From this point of view, the most common and widespread trigger we know is the GSM SIM: a universally accepted and trusted device.

Taken in itself, the GSM SIM is rather "autistic" (or too specialized) and would need specific adaptors to reach terminals other than mobile phones. However, the SIM could use mobile phone's capabilities to transfer data related to its role of Simplicity Device: Bluetooth, IR or even Internet connections (via GSM/GPRS/UMTS).

Thus, the short term implementation of the Simplicity Device could consist of a suitably adapted GSM SIM, installed in a phone equipped with Bluetooth/IR (though in some contexts, this solution could be replaced by/combined with other implementations).

Bluetooth/IR allows the SIM to interact with other devices (e.g., PCs). In addition, a smart phone can provide additional processing power, current SIMs are Java-enabled, typical SIMs contain 64K memory and 128K SIMs are already on the market. Hence a SIM+smart phone is not very far from the ideal characteristics of the storage and processing device outlined above.

Probably, the strongest point of this solution is that it is very good from the point of view of user acceptance. Users already trust their SIM to store sensitive identity and charging data and are used to carrying their mobile phones wherever they go; there is thus no need for them to carry additional devices. Finally, mobile operators could be interested in selling or offering Simplicity in their SIMs, complementing their current services.

With the SIM + Smart Phone as a SD we can also imagine complementary solutions, in which the GSM SIM provides also access to storage on a USB stick (carried by the user) or on the network. All that the user would have to do to "bootstrap" Simplicity, would be to walk near a PC with a suitably equipped phone (and perhaps plug in a USB stick, if a large amount of personal data is needed as well).

Figure 1 depicts the overall Simplicity scenario.

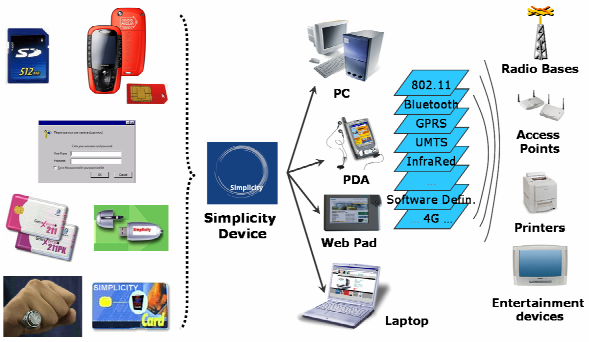


Figure 1. Reference scenario

4. A sketch of the system architecture

A key attribute of Simplicity is re-configurability, at various levels. Re-configurability is typically understood as operating at lower layers (e.g., software defined radio, which indeed may be an "add on" to our solution). However, to integrate different paradigms from the user point of view, it is necessary to break, as much as possible, logical wires that still tie mobile users to networks and services, also at upper layers. This way, heterogeneous and mobile access networks can be really integrated, as IP has glued heterogeneous networks. To enable this full-spectrum re-configurability, our system encompasses three main components: the Simplicity Device, the Terminal Broker and the Network Broker (see Figure 2).

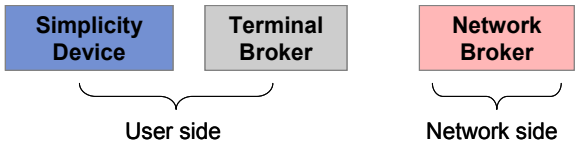


Figure 2. System components

The role of the Simplicity Device, as discussed above, is to store user's profiles, preferences and policies. It also stores and allows the enforcement of user-personalized mechanisms to exploit service fruition, to drive automatic adaptation to terminal capabilities, and to facilitate service adaptation to various network technologies and related capabilities.

The Terminal Broker is the entity that manages the interaction between the information stored in the SD and the terminal in which the SD is plugged in. The Terminal Broker enables the SD to perform actions like terminal capability discovery, adaptation to networking capabilities and to the ambient, service discovery and usage, adaptation of services to terminal features and capabilities. The Terminal Broker caters also for the user interaction with the overall Simplicity system (including network technologies and capabilities).

The Network Broker has the goal to provide support for service description, advertisement and discovery. Moreover, it orchestrates service operation among distributed networked objects, taking into account issues related to the simultaneous access of several users to the same resources, services, and locations. It also shares/allocates available resources, and manages value-added networking functionality, such as service level differentiation and quality of service, location-context awareness, and mobility support.

Summing up, our system results from the combination of the Simplicity Device with a Brokerage Framework (that in turn includes Terminal Broker and Network Broker). The Brokerage Framework will use policy-based technologies (e.g., policies for mobility support, QoS, security, SW downloads) to orchestrate and adapt network capabilities, taking account of user preferences and terminal characteristics.

In this vision, all complex tasks are handled by a middleware that adapts communication technologies and service platforms to user service needs, limiting user interaction to special decisions. Users could roam across environments with different "degrees of intelligence", from "bare" communications services to highly sophisticated smart spaces providing value added context-aware services, while automatically triggering home/building/public-space functionalities.

The final result is that the middleware level manages all the complexity, while users enjoy simple fruition of services.

To define the system architecture the project adopted the following approach. The system requirements and functionalities were derived from about 30 user scenarios produced by the Simplicity partners. A process of selection, refinement and merging led to the creation of four "reference scenarios", which provide an adequate illustration of the way in which Simplicity is designed to be used [2] [6].

The first user scenario ("Mobile Worker"), describes how Simplicity can help a business manager in his work. "Media Streaming & HES (Home Environment Server)" presents a user scenario in which the user uses Simplicity to buy, download, and consume audio and video. "Buy and Use a Self Learning Simplicity Device" describes how Simplicity monitors and learns from user behaviour. The last, "Tour Guide", scenario illustrates how a Simplicity enabled terminal can "discover" services for the user, based on a user profile [6].

All functionalities identified in the original 30 some scenarios are present in at least one of the "reference scenarios" [2].

Once fully defined, the "reference scenarios" were formalized as UML use case and sequence diagrams. By analyzing these diagrams it was possible to identify about 20 main functionalities to be provided by Simplicity, as well as a number of sub functionalities. In addition, the study identified a number of specific requirements on Simplicity sub-systems: the Simplicity Device, the Terminal Broker and the Network Broker. Requirements were identified also for user interaction with the Simplicity system. A preliminary investigation identified tentative "economic requirements" for the system [2]. Collectively these requirements provided input to the Simplicity System architecture design [4].

In parallel with the development of system requirements, the project developed a first tentative Business Model for Simplicity [2]. The model identifies the products and services expected to generate revenue in the Simplicity value chain. It suggests that network operators could play a key role in the chain – but also identifies a number of other actors: distributors, value added service providers, trust institutions, Simplicity Applications Developers manufacturers, and, of course, the owners of Simplicity Intellectual Property. The document on the business model briefly discusses also a possible extension to include "truly transparent roaming" by business and leisure travellers. This possibility, it suggests, could lead to a significant expansion of current markets for mobile data traffic, benefiting all actors in the value chain.

5. Conclusions and future research directions

The Simplicity project addresses a crucial issue for future systems beyond 3G (and for the Internet at large). The concept developed in the project can directly impact the way citizens live and work. We intend to prove this concept by implementing our proposed architecture, to show its feasibility. A key parameter to judge the outcomes of Simplicity is the user acceptability and usability of the Simplicity Device. Proof of this will be shown via a user-centred approach. This concept can also be instrumental in opening up new research directions (or extensions of current ones), including:

- User Profile definition and handling;
- User (and SD) tailored applications and API;
- Middleware tools for high layer re-configurability;
- Network planning and dimensioning as a function of customers profiles (stored in the SD);
- Services and resources (including access networks) discovery and selection, as a function of SD profiles and of current network status;
- Dynamic network (auto)configuration and resource allocation as a function of users' profiles, collected users' statistics, observed current users' behaviour, users' location, and current network status;
- Flexible authentication mechanisms;
- Seamless service accessibility through heterogeneous and independently-owned network infrastructures;
- Simplicity aware network management, and related policies (proactive and reactive).

6. Acknowledgements

This work has been partially funded by the European Union in the framework of the project IST-2004-507558 Simplicity.

The contribution of all Simplicity partners to this paper is gratefully acknowledged. However, that support should not be construed as an endorsement of the views, results and conclusions contained in this paper, for which the author is wholly responsible.

7. (Meta)References

[1] Website of the Simplicity project: http://www.ist-simplicity.org

[2] Simplicity Deliverable D2101: "Use cases, requirements and business models"; http://server.ist-simplicity.org/deliverables.php

[3] N. Blefari-Melazzi, G. Ceneri, G. Cortese, N. Davies, N. Dellas, A. Friday, J. Hamard, E. Koutsoloukas, C. Niedermeier, C. Noda, J. Papanis, C. Petrioli, E. Rukzio, O. Storz, J. Urban: "The Simplicity Project: easing the burden of using complex and heterogeneous ICT devices and services. Part II: State of the Art of Related Technologies", IST Mobile&Wireless Communications Summit 2004, June 27-30 2004, Lyon, France.

[4] Simplicity Deliverable D2201: "Initial system architecture specification"; http://server.ist-simplicity.org/deliverables.php

[5] N. Blefari-Melazzi, G. Bianchi, G. Ceneri, G. Cortese, S. Kapellaki, K. Kawamura, C. Noda, S. Salsano, I. S. Venieris: "The Simplicity project: easing the burden of using complex and heterogeneous ICT devices and services. Part I: Overall Architecture", IST Mobile&Wireless Communications Summit 2004, June 27-30 2004, Lyon, France.

[6] R. Seidl, F. Berger, S. Kapellaki, T. Frantti, E. Rukzio, J. Hamard, N. Blefari Melazzi: "User scenarios for simplified communication spaces", IST Mobile&Wireless Communications Summit 2004, June 27-30 2004, Lyon, France.