



Serenoa



Serenoa is aimed at developing a novel, open platform for enabling the creation of context-sensitive service front-ends (SFEs). A context-sensitive SFE provides a user interface (UI) that exhibits some capability to be aware of the context and to react to changes of this context in a continuous way. As a result such a UI will be adapted to a person's devices, tasks, preferences, and abilities, thus improving people's satisfaction and performance compared to traditional SFEs based on manually designed UIs.

At a glance

Project title (full):

Multidimensional, context-aware adaptation of Service Front->Ends (STREP)

Project coordinator:

Telefónica I+D (ES)

Partners:

Université Catholique de Louvain (BE),
CNR-ISTI (IT),
SAP AG (DE),
GEIE ERCIM (FR),
Fundación CTIC (ES),
W4 (FR)

Duration:

September 2010 – August 2013

Total cost:

€ 5.1 M

Website:

<http://www.serenoa-fp7.eu>

The Problem

Today's Service Front-Ends (SFEs) are often designed with the assumption that they are going to be used in a single, static, predefined context of use. Such a context usually involves an able-bodied user who is using a typical set of input and output devices (e.g., a desktop / laptop computer) and who is sitting in a stable environment (e.g., home, office).

Any deviation from this assumption (for instance, hand tremor due to aging or low vision for a user, switching from one device to another such as from a feature phone to a multi-touch smartphone, riding on the noisy environment of a jostling bus) may significantly hamper users' effectiveness – not because of any inherent barrier to interaction, but because of a mismatch between the effective context(s) of use (and its evolution over time and space) and the static, predefined context of use.

The *context of use* is a dynamic, structured information space that includes a model of the user (who is intended to use or is actually using the system), the hardware-software platform (which includes the set of computing, sensing, communication, and interaction resources that bind together the physical environment with the digital world), and the social and physical environment (where the interaction is actually taking place).

Our Approach

We argue that SFEs should be automatically adapted to better suit disparate and evolving contexts of use while involving the end user in two major ways: *the user can intervene in the adaptation* (e.g., by controlling, suggesting, accepting/rejecting adaptations, requesting better adaptations) and *the system can learn from the user* (e.g., by observation, by sensing, by machine learning). Sophisticated adaptations can include the re-distribution, migration or inter-modal remoulding of user interfaces.

The figure below gives an overview of the Serenoa's overall approach. We propose to conduct research and development work under two domains: the *computational framework domain* and the *technology domain*. The former is devoted to devise the main scientific principles and formalisms. The latter is concerned with the development of the enabler technologies, their runtime support and tools.

The computational framework will be constituted by a Reference Framework (CARF), a Design Space (CADS) and the CARFO Ontology, which will be the main formalism supporting the representation of adaptation knowledge.

The technology domain will be concerned with the enabling technologies (*meta-models, languages, runtime*) and an agile methodology for the development of context-sensitive SFEs, in accordance with the CARF and CADS.

Guiding Principles

The development of the Serenoa project will be driven by the following principles:

- New concepts, languages, (intelligent) runtimes and tools are needed to support multi-dimensional context-aware adaptation of SFEs. These artefacts will enable SFE engineers to concentrate on the functionality rather than on the implementation details concerning the adaptation to the multiple dimensions of the context of use.
- Keeping Humans in the Loop. This principle is twofold. On the one hand, end users should be able to provide feedback or even guide the adaptation process according to their preferences or previous experiences with the system. On the other hand, authors, developers and engineers should be able to guide the adaptation process according to their experience and domain knowledge.
- Open Adaptiveness. A system is open adaptive “if new adaptation plans can be introduced during runtime”. Adaptation in ubiquitous computing environments (such as in ambient spaces) is also necessary in order to deal with multiple devices, interaction resources and modalities.
- Covering the full adaptation lifecycle to support a full adaptation life-cycle that will result into feedback loops (coming from end users) in order to inform any future adaptation.

