# Serenoa Newsletter

## Multidimensional Context-Aware Adaptation of Service Front-ends

March 2012 / n. 3

### Serenoa News

The FP7 Serenoa project is now entering its second and decisive half with important intermediate results and challenging research questions to be addressed in the coming months.

The period since the last newsletter has been one of consolidation and design for the future: theoretical building blocks that were produced in the first year have been packaged as software modules, which together serve our main goal of context-oriented adaptation of SFEs.

Along with the publication of the newsletter, we are providing the first versions of essential Serenoa components, including its Adaptation Engine, Context Management infrastructure and Runtimes. In addition, new useful tools for the definition of adaptation concepts have been delivered, including an Adaptation Rule Language and a first version of the Knowledge Base built upon the CARFO ontology.

Serenoa also enters now in the critical stage for its standardization objectives: a W3C Working Group on Model-Based UIs has been established and has started collaborating with the consortium.

During the past period, three new publications were successfully approved, namely: 'Past, present, and future of model-based user interface development', by Paternò, Meixner and Vanderdonckt; 'Cross modality adaptation of service front ends', by Paternò, Sisti, and Spano; and 'Automatic Reverse Engineering of Interactive Dynamic Web Applications to support Adaptation across Platforms', by Bellucci, Ghianni, Paternò, and Porta.

Other scientific articles were jointly submitted in 2012, and hopefully they will be announced in our next newsletter.

In this newsletter you find further information about our communication channels and achievements. In case of suggestions, comments or doubts, please contact us at: serenoa@tid.es

### Serenoa Roadmap

Two versions of the Serenoa roadmap have been released up to now. In the 1st version we provide insights on the current relevant scientific and technological challenges, as well as on the vision, perspective and contribution of each project partner. In the 2nd release we address different issues: the goals/scope of the project and the rationale for decisions; the relationships between the WPs and how they contribute towards the goals of the project; the project results and how they will be made available and supported; the role of the authoring tools and their fit in the project strategy; the role of the applications in demonstrating the overall project goals.

### Events

**ACM SIGCHI Symposium on Engineering Interactive Computing Systems**

**Date**: *June 25-28, 2012*

**Chairs**: *Simone Barbosa (PUC-Rio, Brazil) and José Campos (University of Minho, Portugal)*

The fourth EICS will be held on IT University of Copenhagen, Denmark. EICS is devoted to engineering usable and effective interactive computing systems. Topics of interest include multi-device interactive systems, new and emerging modalities (e.g., gesture), entertaining applications (e.g., mobile and ubiquitous games), safety critical systems (e.g. medical devices), and design and development methods (e.g., extreme programming).

**Webpage:** eics-conference.org/2012

## Requirements

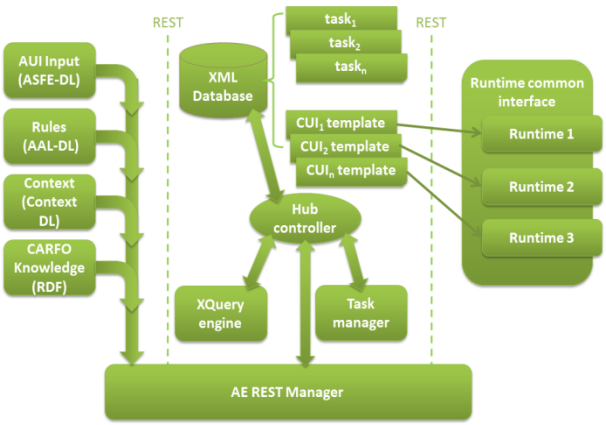
| | Not to be implemented | To be implemented | Implemented | No evaluation | To be evaluated |
|---|---|---|---|---|---|
| Accessibility | | X | X | | X |
| Control over the adaptation process | | X | | | X |
| Multimodality | | X | | | X |
| Cross-platform consistency | | X | X | | X |
| System and task continuity | | X | | | X |
| User-dependent adaption | | X | | | X |
| Intuitiveness | | | X | | X |
| Satisfaction | | X | | | X |
| Visual feedback | | | X | X | |
| Customization | | X | | | X |

The goal of a requirement analysis is to design a system for the user's needs, wishes and capabilities. In Serenoa, requirements were gathered in the very first step and continuously lead following activities such as writing scenarios and conducting other usability engineering methods to specify and evaluate the system and its behavior.
The first list of requirements is now being refined and updated and the evaluation of the latest requirements implementation is ongoing.

## Theoretical Models


The theoretical models of Serenoa, CADS and CARF, aim at guiding developers and designers during the complete SDLC, i.e. by listing alternative possibilities to implement context-aware adaptation, and also by permitting the analysis and comparison of adaptive and adaptable applications. Both models have been progressing along the project development and new versions are presented and described in deliverable 2.1.2.
The updated version of the CADS is illustrated by the figure on the left.

## Adaptation Engine




The Serenoa Adaptation Engine is where abstract UIs and user and developer rules are processed together to produce concrete SFEs adapted to the context of use. The engine, built upon distributed engineering principles, calculates an optimal transformation between abstract and concrete levels and afterwards uses XQuery and XSLT to perform transformations and handles its results to the runtime for further adjustments or direct presentation to the end user.
A block diagram of the engine is presented on the figure on the left.

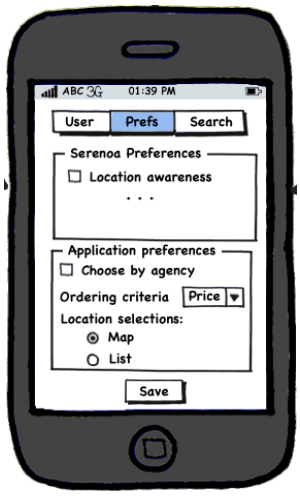
## CARFO


CARFO is the Context-Aware Reference Framework Ontology that is being developed within Serenoa. The CARFO Ontology is based in the theoretical models already proposed by the project, namely CARF (illustrated by the figure on the left) and CADS. Some of the different branches defined in the CARF model and dimensions defined in CADS would be partially integrated when modeling CARFO. But at the same time, it is reusing some of the relevant previous work gathering context information already done by some popular ontologies and vocabularies.

## Demonstrator

The M18 demonstrator is a scaled down prototype of the Serenoa framework in which we showcase the project concepts at large with the first versions of the key components rather than with distributable components, which is targeted for month 24. We do not intend to test the full functionality, but the soundness of the concept and the technology we are building.

For that purpose, we have decided not to implement one of the scenarios proposed in the past documents in the project because it would be too complex and distract us from the main focus.



Preferences

The demo features a front-end described in abstract terms in the project's own ASFE-DL language. This abstract description is adapted using context information (coming from the Context Manager) and rules coded in AAL-DL, and then rendered in two of the multiple possible runtimes. On the one hand, a mobile application (pictured) using IDEAL2 and SCXML as its Concrete UI (CUI) and different Web markup languages as its Final UI (FUI) description language has been developed. On the other hand, a speech browsing prototype has been built by using MARIA XML as its CUI and VoiceXML as its FUI.

We expect to integrate some of the aspects of the demonstration with the desktop Authoring Tool to illustrate how aspects of the design will be changeable in near real time and to check their impact on the final UI.

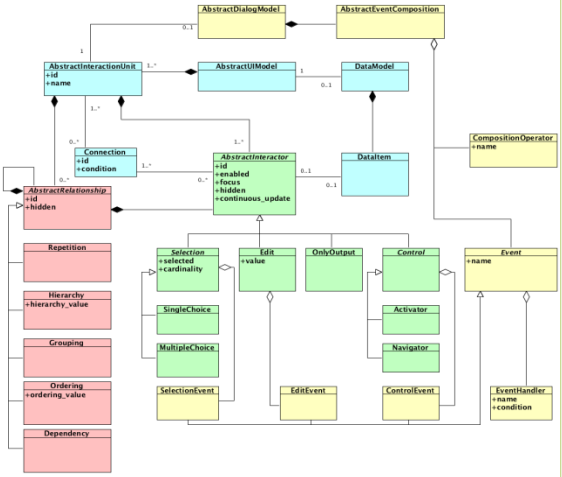
The demo has been developed using the 'glassbox paradigm' so the internal activity can be easily monitored for demonstrative purposes.

We intend to deliver the demo on time for Serenoa's M18 review around mid April, including a video version for external dissemination.

## Languages

### ASFE-DL

The Advanced Service Front-End Description Language (ASFE-DL) is aimed at enabling the development and authoring of context-aware SFEs. The interfaces modelled through such language will be adapted to the context by exploiting the rules defined through the AAL-DL (described below). By leveraging on past experience of UI languages that members of our consortium have already authored or co-authored, we plan to build a more complete language, unifying concepts and adding new features that will allow ASFE-DL to meet the Serenoa requirements and to go beyond the state of the art in this field. In this first version of the ASFE-DL, we focused on the Abstract UI level (see figure on the right). The ASFE-DL is currently being submitted to a standardization plan to W3C.



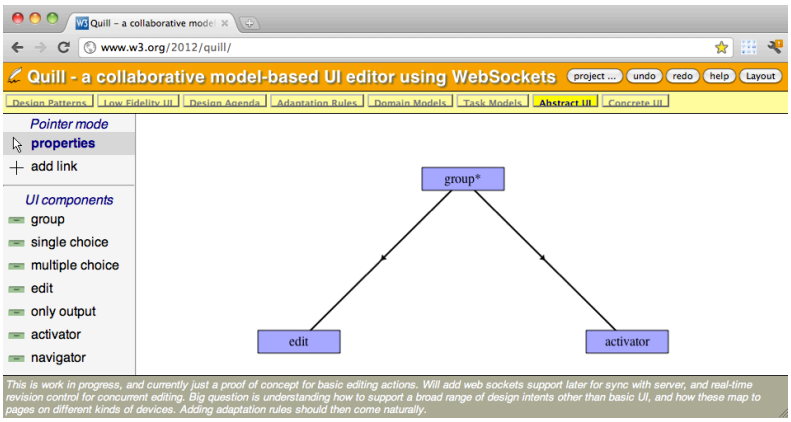

### AAL-DL

A first version of the Advanced Adaptation Logic Description Language (AAL-DL) has been already released in Serenoa. The AAL-DL is a high-level language intended to express advanced adaptation logic in a declarative manner. The basic idea is that the UIs modelled through ASFE-DL (described above) will be adapted to the context by exploiting the rules defined through the AAL-DL. Such rules have been expressed through an ECA-based (Event, Condition, Action) format where: *events* are changes that can occur in the context state or in the UI state; *conditions* are Boolean predicates referring to context state or UI state; *actions* are changes affecting the interactive application.

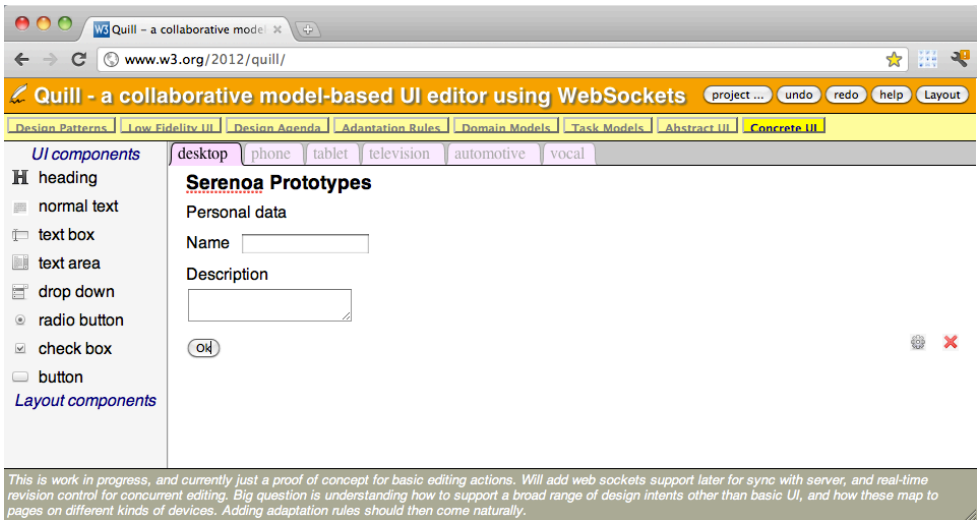
## Follow us

## Quill

Quill is an authoring tool for distributed editing of model-based user interface designs, that runs within recent web browsers and will be used to edit the Serenoa adaptation rules. The HTML5 CANVAS element is used for graphical models, and WebSockets for communication with the server. Quill organizes work into projects, and holds the project data on a Web server. Multiple people can work concurrently on the same project, in an analogous fashion to Google Docs. Each person can only view and edit one layer of the Cameleon reference framework at a time. A novel feature of Quill is the ability to work top-down or bottom-up. This makes use of a rule engine that runs on the server, and which responds dynamically to the changes authors make to user interface models. As an example, an author could choose to work at the concrete U layer on models for a desktop interface. The rule engine will then automatically update the models at the abstract UI layer, and from there propagate design changes to models at the concrete UI layer for other chosen target platforms such as mobile or TV. The adaptation rules can add tasks to Quill's Design Agenda, for the authors to deal with. This is necessary when the adaptation process requires human intervention. Quill has a modular design that separates out the authoring user interface for each layer, and makes it easy to switch the model visualization used at each layer. Layout algorithms are used to adapt the visualization to the author's browser window size, and authors working on the same project will see their own visualization.




These two snapshots of Quill, illustrate task model and CUI edition. The menu on the left allows the dragging-and-dropping of the UI elements of interest. These elements can be edited according to the developers' needs. The multiple tabs available permit considering different context of use, and granularity levels.




Quill is work in progress. The current focus is on developing the authoring user interface, and exploring the role of force directed animated layout techniques. A graph visualization has been developed for the abstract UI, and plans are in place for adding an alternative visualization involving nested containers. We plan to track the emerging model specification languages from the W3C Model-Based UI Working Group. Work has just started on the server-side adaptation engine where we are considering using the JESS forward chaining rule engine. A snapshot of the Quill demo can be found at http://www.w3.org/2012/quill/

## Follow us

## Announcements

### First face to face Meeting

The first face to face meeting of the Model-based User Interface Working Group of W3C took place on February 9th and 10th at DFKI venues in Kaiserslautern (Germany), more than 20 participants were present to collaborate with the discussions about model-based approaches for developing user interfaces.

At first, the participants discussed the benefits and shortcomings of adopting model-based approaches. As *benefits*, the group emphasized: the separation of design concerns, a better documentation of the UI design, the easier adoption of design guidelines, less efforts to target multiple platforms, the possibility of choosing the development path (i.e. top-down or bottom-up). As *shortcomings*, were highlighted: the higher learning curve, additional efforts for the development, difficulty in generating the transformations between levels and the absence of consent regarding the terminology used and the scope of each abstraction level.

The working group received 8 submission proposals which will be analyzed until September 2013, when a candidate recommendation will be selected. Each proposal was presented by its authors during the meeting.

The second day was dedicated to discuss main concepts about the *Task and Domain* level, and for the *Abstract User Interface* level. The discussion goals included: comparing different languages, identifying their commonalities and differences, and based on the common concepts try to reach a standard terminology for each level. For the discussion, the participants were divided in two task forces.

- For the **Task and Domain** level, the participants defined some concepts, e.g. relationships, structures, operators, categories, objects, conditions, importance/priorities, frequency, state, collaboration, evolution;
- For the **AUI** level, the participants compared the following languages: UseML, IdealXML, CUI DSL (University of Dresden) and MariaXML.

Aiming to progress the work of the MBUI Working Group there are weekly calls planned and further face to face meetings.

This working group is chaired by Gerrit Meixner and its contact person is Dave Raggett.

Further information can be accessed online at: http://www.w3.org/2011/01/mbui-wg-charter

**MBUI Working Group participants at the Living Lab – Smart Factory, in DFKI - Kaiserslautern (Germany)**

*Serenoa project is funded by the European Union through its Seventh Framework Programme as a STREP Project nº FP7-ICT-258030 from September 2010 to September 2013*