



Multi-Dimensional Context-Aware Adaptation of Service Front-Ends

PROJECT FINAL PUBLISHABLE REPORT

Grant Agreement number: 258030

Project acronym: SERENOA

Project title: Multi-Dimensional Context-Aware Adaptation of Service Front-Ends

Funding Scheme: STREP

Period covered: from 1st August 2010 to 30th September 2013

Name of the scientific representative of the project's coordinator, Title and Organization:

Dr. Javier Caminero Gil

Telefónica Investigación y Desarrollo, fjcg@tid.es

Ronda de la Comunicación s/n

E-28050 Madrid

Spain

Tel: +34-914832671

E-mail: fjcg@tid.es

Project website address: <http://www.serenoa-fp7.eu>

The SERENOA Consortium

- TELEFÓNICA INVESTIGACIÓN Y DESARROLLO, <http://www.tid.es>
- UNIVERSITÉ CATHOLIQUE DE LOUVAIN, <http://www.uclouvain.be>
- ISTI, <http://giove.isti.cnr.it>
- SAP AG, <http://www.sap.com>
- GEIE ERCIM, <http://www.ercim.eu>
- W4, <http://www.w4global.com>
- FUNDACION CTIC <http://www.fundacionctic.org>

Table of Contents

| | | |
|----------|--|-----------|
| 1 | Final Publishable Summary Report | 4 |
| 1.1 | Executive Summary | 4 |
| 1.2 | Project context | 5 |
| 1.3 | Main S&T results and foreground | 6 |
| 1.3.1 | <i>Description of SERENOA approach</i> | 6 |
| 1.3.2 | <i>Authoring Tools</i> | 6 |
| 1.3.3 | <i>Languages</i> | 10 |
| 1.3.2 | <i>The Advanced Adaptation Logic Description Language (AAL-DL)</i> | 11 |
| 1.3.3 | <i>Theoretical framework</i> | 12 |
| 1.3.4 | <i>Run-time components</i> | 15 |
| 1.3.5 | <i>Applications prototypes</i> | 18 |
| 1.3.6 | <i>Main S&T Results</i> | 22 |
| 1.4 | Potential impact & main dissemination activities | 25 |
| 1.4.1 | <i>Added value</i> | 25 |
| 1.4.2 | <i>Impact</i> | 25 |
| 1.4.3 | <i>Serenoa beneficiaries</i> | 26 |
| 1.4.4 | <i>Industrial Advisory Board</i> | 27 |
| 1.4.5 | <i>Main dissemination Activities</i> | 28 |
| 1.5 | Project public website and contact details | 31 |
| 2 | Use and dissemination of foreground | 33 |
| 2.1 | Section A | 33 |

1 Final Publishable Summary Report

1.1 Executive Summary

While most of the existing applications still target at a pre-defined context of use, of an able-bodied user, with a Desktop PC interacting in a stable environment, currently, the end users interact with different devices, using multiple modalities and from varied situations (e.g. while cycling, driving, shopping). Although the contexts significantly vary, stakeholders, including developers and designers, are not able to find support that is complete enough to aid the generation of applications that are suitable for end users to interact in their environments. As a result the development process is delayed and complex, and the applications launched often target at a specific context of use. Serenoa aims at extending the state-of-the-art of adaptation, by providing support for stakeholders to develop applications that can successfully adapt their front-ends according to the characteristics and constraints posed by multiple contexts.

To provide support for adaptation, Serenoa achievements encompass a series of tools and methods, including: reference models, languages, authoring environments, prototypes, evaluation criteria, frameworks and toolkits. With such tools and methods, Serenoa provides an innovative platform to support the development of applications that consider adaptation. By using the outcomes provided by the project, stakeholders (as developers and designers) can find guidance to better define, develop, implement, evaluate, and maintain their applications, facilitating and accelerating the development of the UI adaptations.

Multiple application domains can benefit from adaptation and from Serenoa outcomes, including applications targeting at: mobile terminals, smart phones, tablet PC's, and web-based devices. The contexts considered also vary, for instance concerning users with different profiles, preferences and impairments, devices with different modalities, dimensions and capabilities, and environments with different locations, noise and light levels. For example: the user interface can be adapted in a warehouse picking scenario, in which the end user interact with a head mounted display and vocal modality; or in a medical scenario, in which an avatar can be used to mediate the end user interaction, using a simplified language to improve the accessibility of the system.

Serenoa has a threefold impact: not only the industry and the academia can benefit from the outcomes of the project, but also the standardization bodies. Industry can use the authoring environments, apply the frameworks and use the toolkits produced within the project. Academia and researchers can benefit from the reference models, languages and problem spaces defined within Serenoa. The standardization bodies can benefit from a unified terminology, a standard approach and also find progress in the definitions of model-based concepts for the design of user interfaces.

The results produced by Serenoa provide benefits for both end users and stakeholders (including UI designers, software engineers and architects, developers, and project managers). For the end users, the user interfaces that are able to adapt themselves according to the context, tend to provide higher quality levels, i.e. enhancing the usability and the accessibility levels, improving the performance and the user satisfaction. For stakeholders, support can be found during the whole SDLC (Software Development Life Cycle), since the definition and design process, through development and implementation, until the evaluation and maintenance. As a consequence, the context-aware applications generated with Serenoa support tend to have a shorter time-to-market, a better cross-device consistency and an easier interaction.

1.2 Project context

Even though software development environments, standards and engineering methodologies have significantly evolved and matured over the last decade, designing and implementing context-aware application user interfaces is still complex, and therefore expensive.

Businesses are seeing increasing diversity in the kinds of devices being used (i.e. workstations, desktops, laptops, smart phones, tablets, embedded devices, etc.), but this is increasing the cost of developing and maintaining user interface designs. Aligning business and Information Technology (IT) is a constant challenge, not only because of the constantly evolving standards and underlying technologies, but also because of the accelerating pace at which functional requirements change throughout the lifetime of applications. Business applications need to adjust rapidly to fulfill changing requirements, and this trend is accelerating as users require more and more mobility.

Second, implementing Human-Computer Interaction (HCI) systems requires anticipating user-driven interaction combinations that are not always easy to anticipate. Adding context management to the HCI domain increases complexity further because it requires taking into account multimodality (i.e. keyboard & mouse, touch screens, voice, gesture, etc.), geographical location, social context, accessibility and user preferences.

Finally, the underlying technologies involved in the development of user interfaces are multiple and keep changing and emerging at a rapid pace. Development therefore often requires technical experts, who are hard to find and retain. As a matter of fact, most of the effort in a traditional project is spent on infrastructure and technical tasks, and not on business issues. This often contributes to frustration of business stakeholders and end-users. There is also a steep learning curve for each new platform that has to be mastered: Android with Java, iOS with ObjectiveC, Windows with C#, Linux with C/C++, the Web with HTML and JavaScript, and so forth. Translating user interface designs into each of these platforms is time-consuming and expensive.

A variety of approaches have evolved in response to the above challenges. These include the Eclipse Integrated Development Environment, the Java Platform Enterprise Edition (Java EE), W3C's HTML5, JavaScript libraries (e.g. jQuery), and OMG's Model-Driven Architecture (MDA). These have been supported by work on software development methodologies, such as Model-Driven Engineering (MDE) or Agile project methodologies, such as Scrum or xP, or with UI design approaches such as User-Centred Design (UCD). When it comes to designing user interfaces, Model-Based approaches bring many benefits by filling the gaps where traditional approaches are inadequate for agile development, and struggle to cope with the reality that requirements are not fully understood at the outset of a project, but rather emerge as designs are tried.

Model based user interface design is about separating out design concerns at different levels of abstraction using both declarative and procedural knowledge. The declarative part can be expressed as task models, abstract user interface (AUI) models and concrete user interface (CUI) models.

Other Model-Based approach benefits include simplification through automation and code generation, a clean separation of concerns, e.g., IT and business issues, and the fact that the model becomes a common language for all project stakeholders and helps IT align with business. People vary in how they approach the challenges of user interface design. We will not realize the true potential of Model-Based user interface design if we only support a top-down methodology. A significant number of people find it easier to start at the concrete level and derive the abstract level as they go. Subsequent revisions to the abstract level should be possible without throwing away the design work done at the concrete level. However, traditional Model-based design also suffers from a lack of attention to adaptation to the context of use, and this is worsened by a lack of standards for expressing adaptation rules.

1.3 Main S&T results and foreground

1.3.1 Description of SERENOA approach

Serenoa's proposed architecture for supporting user interface adaptation is depicted in Figure 1. Additional details are given in the following sections.

Firstly, at design-time, 'Authoring Tools' are used to support the design of model-based user interfaces. The authoring tools support Serenoa's abstract user interface description language, as well as the Serenoa language for expressing adaptation rules (see 'Languages').

Secondly, the 'Theoretical framework' consists of reference models and an ontology. These reference models are aimed at guiding developers and designers during the complete software life-cycle, listing alternative possibilities for implementing context-aware adaptation, and permitting the analysis and comparison of adaptive and adaptable applications. The ontology, based on the reference models, is intended to gather all the knowledge involved in advanced adaptation logic for user interfaces. It is used to inform developers as well as to support adaptation processes in the run-time.

Finally, the run-time phase transforms the description of user interfaces and associated rules into a final user interface implementation. The adaptation engine determines the optimal adaptation for the current context of use, based upon the context models and adaptation rules. To achieve this goal, the context manager provides information related to all the possible contextual dimensions (i.e. the user preferences, the environment, social relationships, etc.). The run-time engine generates the final interactive application according to the context. This module is composed of a set of sub-modules which cover several modalities (i.e. mobile web applications, vocal interfaces, avatar-based interaction or desktop business applications). In fact, different 'Application prototypes' showing the functionality of these sub-modules and the whole Serenoa framework have been developed, see later section for more details.

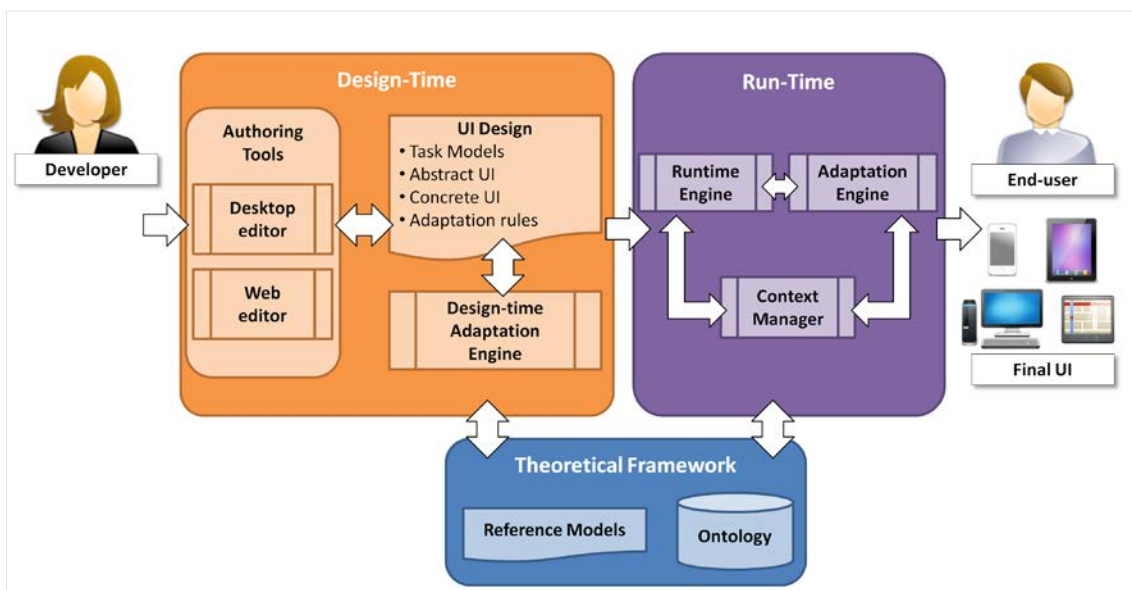


Figure 1. Serenoa's architecture

1.3.2 Authoring Tools

In the Serenoa project, the authoring tools have been developed in such a way that they are easily understandable for non-expert programmers, and fulfill some additional key success factors, such as the usability of the graphical interface, the availability on multiple platforms and the support for concurrent work of several users (engineers and web authors may work as a team) appear as very important requirements. The authoring tools provide support for editing not only the model-based

descriptions of service UIs, but also the context-dependent transformations rules as well. Two types of authoring tools have been developed, namely Eclipse-based plugin and the HTML5 browser-based application. A brief description for each of the tool is given below.

1.3.2.1 Eclipse-based Plugin

The current implementation of the Eclipse plug-in supports the creation and editing of the abstract-level description of a UI model with the help of a *Service UI Editor*, and supports the creation of context-dependent transformation rules with the help of *Rules Editor*.

The *Service UI Editor* is designed as a multi-page editor with *Graphical Editor* and *XML Editor* to facilitate the creation and editing of UI models in both *graphical* and *code* views, as shown in Figure 2 and Figure 3. The *Code View* is developed using the built-in XML Editor from Eclipse in order to display the model. The *Graphical View* is developed using Graphical Editing Framework (GEF), which provides technology to create rich graphical editors and views for the Eclipse Workbench UI. The language schemas and example rule files provided by partners helped in designing the graphical part of the model-based Service UI editor.

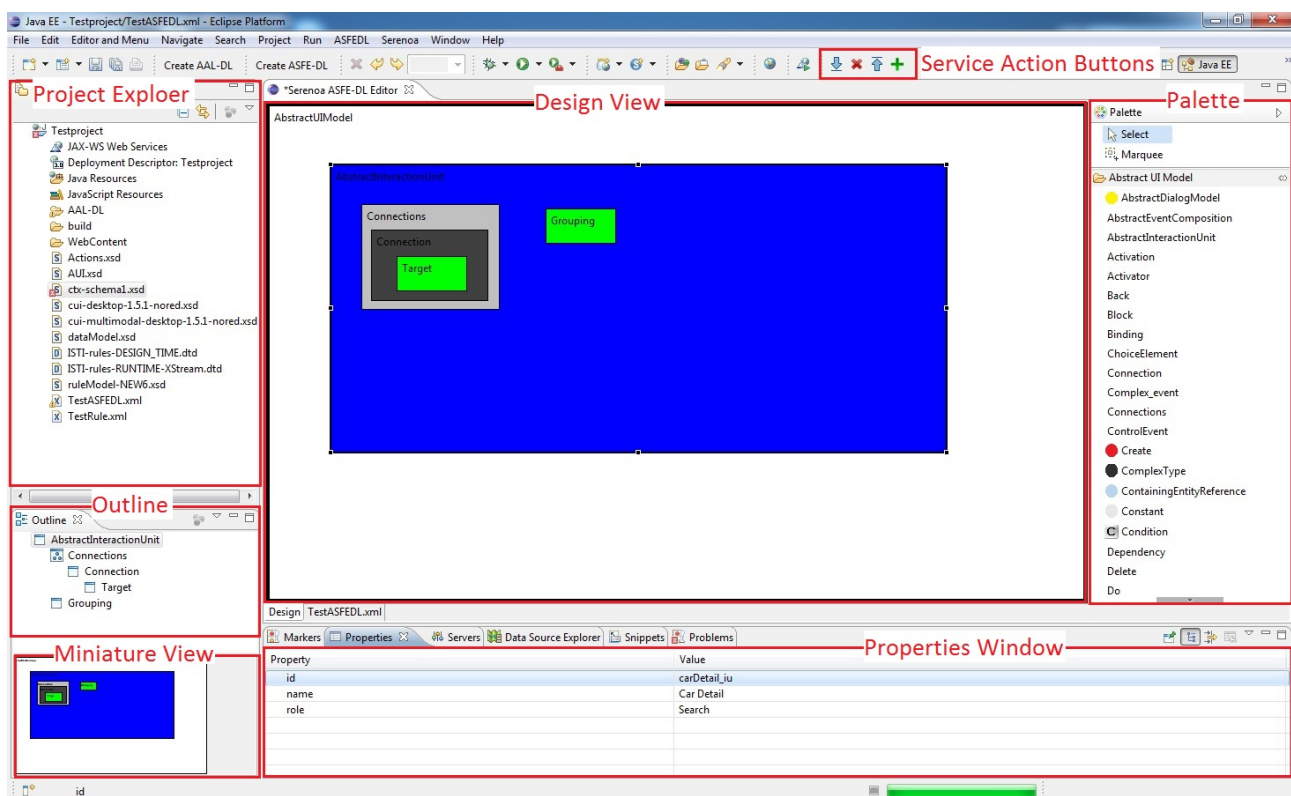


Figure 2. ASFE-DL Service UI Editor – Design View

The graphical part is designed to have a tool palette which contains elements list for creating UI models. The element list is divided into three categories as Connections, Grouping and Events (a structure defined for Abstract Interaction Unit). The drawing area is where to drag and drop elements from the palette, and displays the AIU in a graphical representation (in the form of boxes). This helps the developer to easily and quickly develop the Abstract UI model for Adaptive SFE applications. The editor can also display the Outline view in the form of tree structure and provides displaying and easy navigation feature for the AIU drawn in the drawing area. The property view shows the properties associated with each element that can be set in the fields. The miniature view of the drawing area can be seen giving the idea of graphical part when editing in the XML editor. Both XML editor and Graphical editors are synchronized to reflect changes made at any time in their respective views.

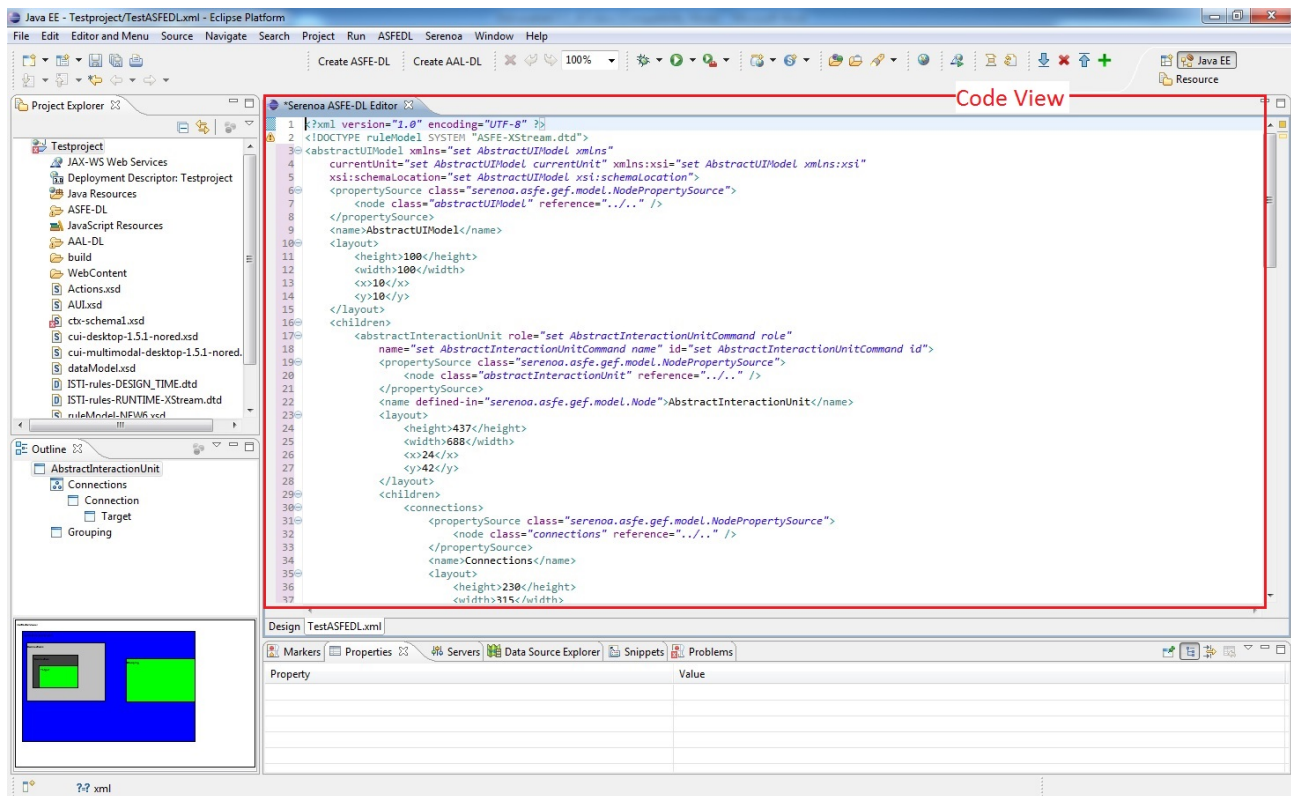


Figure 3. ASFE-DL Service UI Editor – Code View

1.3.2.2 HTML5 Browser-based Application

The second type of authoring tool, named “Quill”, is an experiment that intends to shine a light on opportunities for browser based editing of model-based UI designs by distributed teams of developers. Quill is a work in progress and has been very much a learning process, rather than work on a finished product. Some aspects are better understood than others, and this should be borne in mind when reading the detailed description further. Quill makes use of HTML5 and JavaScript for its user interface. The HTML 2D Canvas is used for realizing graphical models. This allows Quill to run on the wide range of devices that host HTML5 browsers. In practice, Quill has been designed for use on desktop computers with a pointing device such as a track pad or mouse. It has been further adapted to work on tablet computers where the pointing device is your finger.

The Web developer community has largely shifted away from XML to JSON (JavaScript Object Notation) as a lighter weight and more convenient framework for exchange of structured data. Quill follows this trend and explores the use of concise human friendly text-based formats as an alternative to XML. Figure 4 shows the Quill top-level architecture – clients are HTML5 browsers running on desktop computers or on tablets. These are connected to a cloud based server that acts as a collaborative design assistant for the project allowing for distributed teams to work together.

In Quill, the user interface design is held as layered abstractions following the Cameleon Reference Framework¹. The top-most layer contains the domain interfaces and interaction task models. This is followed by models describing abstract user interface in a manner that is essentially independent of the target platforms and modes of interaction. Below this is the concrete user interface, which involves a commitment to particular modes of interaction on broad categories of devices. The

1

<http://giove.isti.cnr.it/projects/comeleon/pdf/CAMELEON20D1.1RefFramework.pdf>

bottom most layer is the final user interface on each of the target devices. This is generated from the concrete user interface where the detailed styling is determined by a theme or skin provided by the project design team. Further information and example languages can be found on W3C Website².

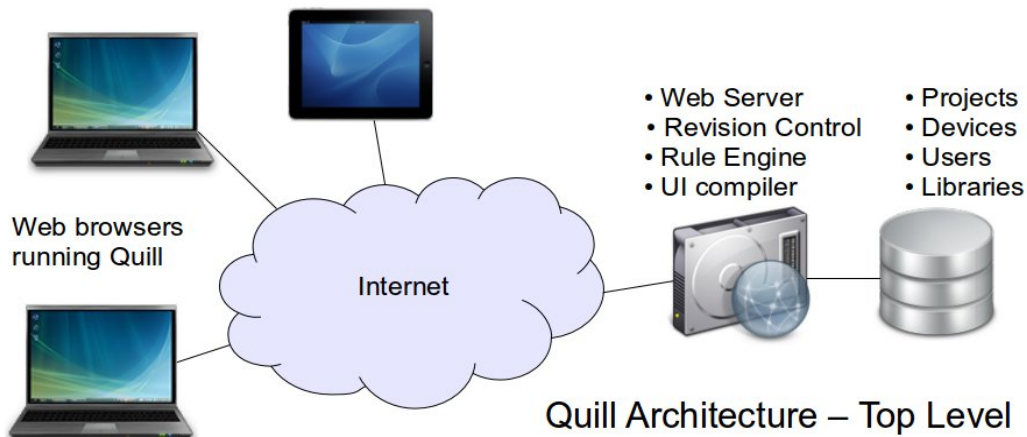


Figure 4. Quill Top Level Architecture

Each project identifies the target platforms, e.g. desktop, smart phone, and tablet. The final UI can be compiled for a specific class of device, e.g. the iPhone. The domain and task models can distinguish between normal and advanced features. Not all tasks are relevant to every platform.

Quill embodies a rule-based design assistant that generates the abstract and concrete user interfaces from the task and domain model. The human designers adjust this design, and Quill then propagates the implications of these changes to search for a consistent overall design for all of the target platforms. In essence, Quill frees designers to make changes without having to do the tedious and error prone work of synchronizing these across the entire project. In principle, Quill could be integrated as part of Web content management systems.

The work is now under way to implement the design assistant in terms of a rule engine that runs on the server. Initial plans for a forward chaining event-condition-action rules were found to be impractical when it comes to bidirectional synchronization, and is now being combined with an approach based upon abduction and condition-relationship rules. This will support synchronization when several users are concurrently editing models at different levels of abstraction. The rules will also be able to critique departures from corporate design guidelines that are intended to ensure that applications have a consistent look and feel across all target platforms.

Quill uses graphical representations for the task and abstract UI models (see Figure 5). This is based upon force directed layout algorithms, where nodes are treated as charged particles that repel each other, whilst arcs are treated as springs. A cone shaped field is used to center the diagram. Automatic layout is used to adapt to the device the model is being viewed upon, as the window and screen size may vary from one user to another.

The force-based layout provides for compact representations that minimize the need for scrolling the view. However, for strictly hierarchical models (e.g. as used for task models), the approach can cause different branches of the tree to overlap, making the diagram harder to understand. To fix

²

<http://www.w3.org/2005/Incubator/model-based-ui/XGR-mbui-20100504/>

this, an investigation was made into the potential for applying nested Voronoi cells. Algorithms for efficiently determining Voronoi cells are available for diagrams with rectangular boundaries, but further work is needed to extend this to support polygonal boundaries. This would allow for a recursive algorithm that starts from the root node and progressively lays out the child nodes using a force directed approach within the polygonal boundary of the Voronoi cells for the parent nodes.

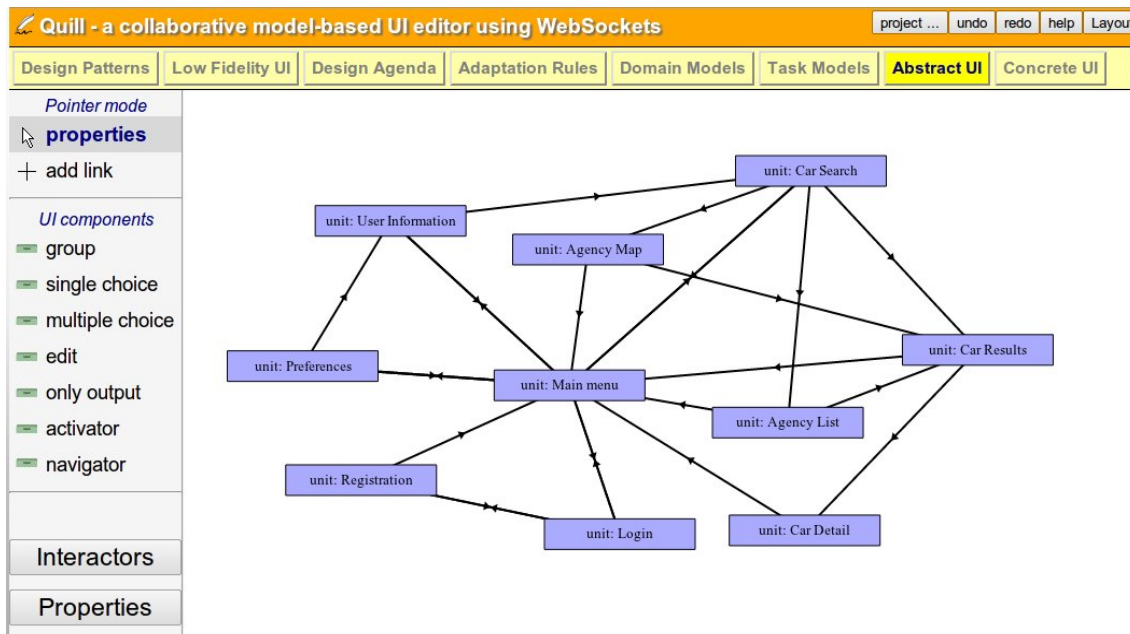


Figure 5. Screenshot showing dialog view of an abstract UI model

1.3.3 Languages

The languages developed in Serenoa cover the specification of adaptive SFEs at different abstraction layers, and the context-dependent transformation rules to be applied on the user interfaces. With the Serenoa solution, the exploitation of both these languages will be supported not only at design time but also at runtime. At design time, the authoring tools help the designers, engineers and web authors to easily create and edit context-sensitive SFEs for different platforms (at both abstract and concrete levels) and relevant context-dependent transformations rules. At runtime, the logical descriptions of the SFEs and of the adaptation rules are transformed in a final, adapted user interface implementation.

1.3.3.1 The Advanced Service Front-End Description Language (ASFE-DL)

The [Advanced Service Front-End Description Language](#) (ASFE-DL) is aimed at enabling the development and authoring of context-aware SFEs. The user interfaces modeled through this language are adapted to the context by exploiting the rules defined through the Advanced Adaptation Logic Description Language (AAL-DL). By leveraging on past expertise on user interface languages that Serenoa members have already authored or co-authored, and on previous experiences they gathered by working in relevant industrial case studies to support requirements of most modern service-based user interfaces, the Serenoa consortium has built a language that meets the Serenoa requirements. The ASFE-DL has been specified at the Abstract user interface level: it describes the UI through a number of abstract interaction units and associated connections in a modality-independent manner. This version has also been submitted as an input to standardization work at W3C. We have then developed two refinements of the abstract ASFE-DL for different interaction modalities (graphical and vocal interaction) to show how it can be

exploited through substantially different ways of interaction. We chose them because the graphical interaction is the most common and the vocal one is acquiring an increasing importance also in mass market products. We have also defined and implemented transformations between ASFE-DL and MARIA, a widely used model-based language for interactive applications.

1.3.2 The Advanced Adaptation Logic Description Language (AAL-DL)

The **Advanced Adaptation Logic Description Language (AAL-DL)** is a high-level language intended to express advanced adaptation logic in a declarative manner. The basic idea is that the user interfaces modeled through ASFE-DL will be adapted to the context by exploiting the rules defined through the AAL-DL. The AAL-DL rules have been expressed through an Event-Condition-Action (ECA)-based format where:

- *Event*: It can be an elementary event or even a composition of events occurring in the interactive application or in the context of use. In any case, its occurrence triggers the application of the rule.
- *Condition* (optional): a Boolean condition to be satisfied to execute the associated action(s). It can be related to something happened before, or some state condition.
- *Action*: How the abstract/concrete/implementation description of the interactive application should change in order to perform the requested adaptation.

In the current specification of the AAL-DL we have considered the definition of first-order adaptation rules (simple adaptation rules like e.g., adapt this service front-end for this platform) and second-order adaptation rules (those that govern the application of adaptation rules by e.g. selecting first-order rules: the action part of a rule can be in turn another rule).

The last version of the AAL-DL is based on the extensive use done in the project of this language to specify adaptation rules. The new features introduced are:

- An “*else*” branch has been added to each ECA rule to better express the actions to do when the associated condition is not verified. It is worth pointing out that this change does not add much expressivity power to the language, but it aims at making the AAL-DL rules more structured and readable.
- *New operators to compose events* are also added. At the beginning, it was possible to compose events only through Boolean operators: AND (two events both occur in any order), OR (at least one between two events occurs), XOR (only one between two events occurs), NOT (an event does not occur). However, we realized that further temporal combinations of events could be useful, for instance that two events (or two expressions of events) occur one after the other (*sequence* operator), or that the same event occurs multiple times (namely: zero or more times, or one or more times, or at least a fixed number of times).
- There are also *new types of actions*. It is possible to specify an *update action* also in “relative” terms. At the beginning, it was only possible to specify an update action in absolute terms (e.g. by specifying the new attribute values of an element). In AAL-DL 2.0, the possibility to specify an update action (targeted to a particular UI element) with respect to/in terms of another UI element is added.
- In the action part of a rule, the possibility to specify a *reference* to a sequence of actions (e.g. like a “procedure”) that are defined somewhere else (e.g.: in another rule) has been added.
- An *alphabet of event types* is specified in order to better define the list of possible events that can be included in an adaptation rule.

1.3.3 Theoretical framework

The Theoretical ground of the Serenoa Project is structured in three main components: a **Context-aware Reference Framework (CARF)**, a **Context-aware Design Space (CADS)** and a **Context-aware Reference Ontology (CARFO)**. The CARF provides to stakeholders the core concepts for defining and implementing adaptive and adaptable systems. The CADS provides means to analyse, evaluate and compare multiple applications regarding their coverage level of adaptation, especially concerning certain specific dimensions (such as: modality types). The CARFO not only formalizes the core concepts defined by CARF and their relationships, but also enables the request and retrieval of relevant information for defining and executing the adaptation process. Both industrial and scientific domains can benefit of these theoretical models, once they provide support for the whole development life-cycle of adaptive and adaptable applications, i.e. design, specification, implementation and evaluation.

1.3.3.1 CARF

The goal of CARF framework is to assist user interface designers and developers of interactive and ubiquitous software applications and to provide a basis for adaptive user interface for creating context sensitive user interfaces.

The Context-aware Reference Framework (CARF) is a reference framework created to list the most relevant concepts for implementing and executing CAA. The CARF is graphically represented by a mind map and composed by seven central branches. The mind-map has been chosen to represent CARF due to several reasons, including:

- it provides a unified view of an extensive list of concepts;
- it organizes and relates the concepts in a structured manner;
- it enables to navigate through the branches accessing contents in depth;
- the radial structure, rather than a hierarchical one, is close to the human approach to associate concepts, facilitating the comprehension and the understanding of the concepts and their relationships.

While the central branches of CARF (i.e., the ones directly connected to the root) present abstract concepts, the external branches (added under the central ones) list the possible instances for these abstract concepts, and thus aid the implementation, execution and analysis of CAA.

The definition of CARF inherits from several works reported in the literature, and its evolution is documented in the public deliverables D2.1.1 and D2.1.2.

To instantiate the CARF the following sentence must be appropriately respected and filled: At <when>, concerning <to_what>, the <who> <where> must <how> the <what> to improve the <why>. For example, in natural language, it could be: At run time concerning the user visual impairments the system at the client must enhance the color contrast to improve (or assure) its accessibility.

The seven central branches of the CARF (see Figure 6) represent, in clockwise sense, what, why, how, to what, who, when, and where dimensions, defined as follows:

- **What:** represents the resource type that is adapted, generally belonging to three main categories: navigational flow, presentation or content. For example: images;
- **Why:** defines the main goals for the adaptation process, which are expressed in terms of software qualities. For example: adaptation can be performed targeting the improvements of the usability level;
- **How:** defines how the adaptation process is performed, by listing possible methods, techniques and strategies for the adaptation. For example, with the adaptation technique of changing the font size;
- **To what:** lists context information that justifies and defines the adaptation process, i.e., usually the application resources are subject to adaptation according to the user, the platform, or the environment. For example: adapting according to color-blind users;

- Who: refers to the actor who triggers and is responsible for each phase of the adaptation process, e.g.: the end user, the system, or a third party. In a mixed approach both end users and system collaborate with the adaptation process;
- When: represents the state in which the adaptation process is performed, i.e., it can occur at design time, run time, compilation time. For example: adaptations performed at run time;
- Where: this branch refers to the 'location' in which the adaptation takes place, i.e., according to the architectural approach adopted it can be at the client, at the proxy, or at the server. For example: adaptation performed at the server side.

These seven branches compose the core of the CARF, and by adding new instances they can be refined, however they are enough complete to cover the most relevant concepts in this domain, being sufficient to comprise and to express all the necessary phases of an entirely CAA process.

The CARF defines the most relevant concepts for CAA and extensively lists and presents the possibilities regarding its implementation and execution. The CARF can be used before the implementation phase of an application, as an extensive catalogue to guide developers in taking design decisions, or after the implementation phase of an application, to analyze and to evaluate the concepts that were considered, aiding also to identify underexplored areas for future extensions.

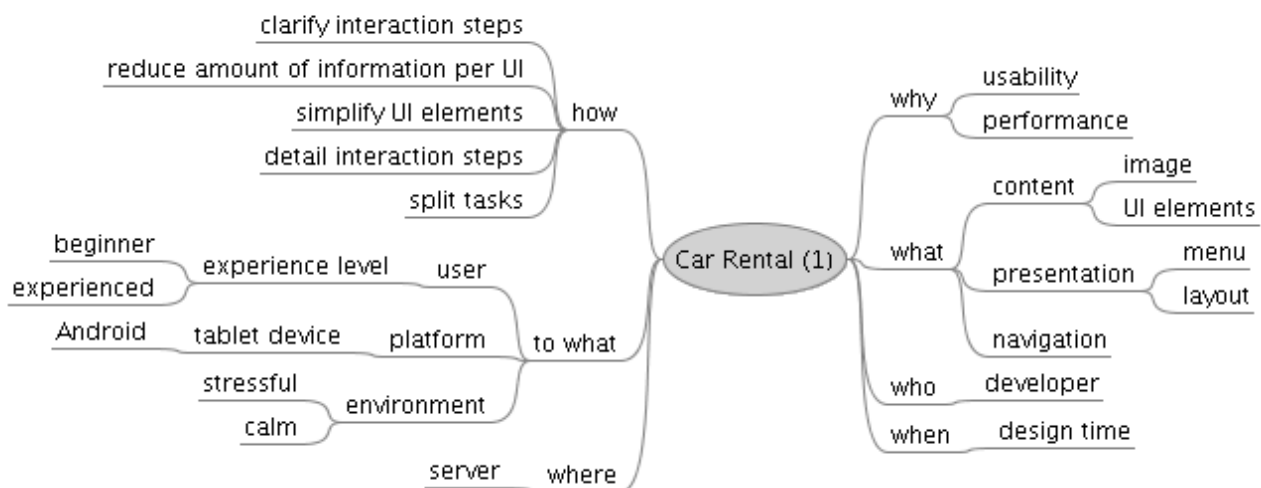


Figure 6. CARF: an example of the context-aware reference framework applied for the Car Rental system

1.3.3.2 CADS

The Context-aware Design Space supports stakeholders in the phases of implementation, analysis, and evaluation of adaptive and adaptable applications. The CADS aids developers before and after the implementation phases. Before it, they can identify possible dimensions and granularity levels for performing adaptation, and after it they are able to analyze, evaluate and compare these dimensions regarding their respective coverage levels. As such the CADS supports the analysis and the comparison of different applications that execute adaptation and during their complete development lifecycle. The CADS has been built in an iterative manner. First, relevant dimensions of CAA were identified based on the SLR. Then the specific granularity levels for each dimension were defined. In a first version of the diagram, due to the fact that not all dimensions represented ordered values, the interpretation of the diagram lead to wrong conclusions. As a result, the diagram was split. Dimensions that are not ordered belong to the CARF (for instance context information). Solely dimensions that have an ordered meaning were kept, for instance regarding the level of applicability of CAA (ranging from the entire application, to specific properties of the UI elements). The evolution of CADS is documented in D2.1.1 and D2.1.2 public deliverables.

The definition of this design space inherited a lot from several work analyzed during the literature review. CADs can be considered as an evolved definition of a problem space.

The CADs, as Figure 7 illustrates, is built as a radar chart, a useful approach to represent multi variable observations with an arbitrary number of variables. Although, in principle this representation is used for ordinal measurements, in the CADs case, qualitative values are represented with their respective empirical scale associated.

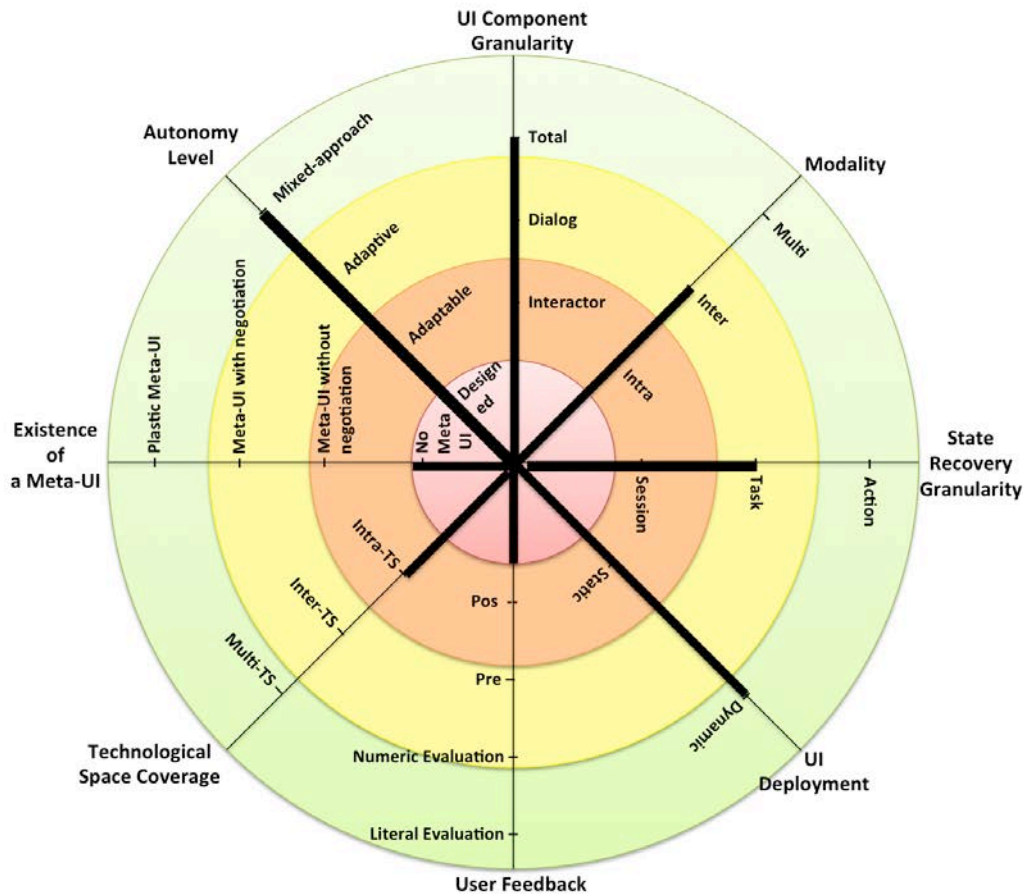


Figure 7. CADs instantiated for an illustrative application

The CADs considers 3 benefits for design spaces, therefore it is comparative because multiple applications can be analyzed according to consistent criteria; exploratory, because each dimensions can be analyzed in terms of exploration, i.e., identifying additional opportunities for extensions; and descriptive, because the dimensions are precisely pre-defined, in a consistent and unique way. Moreover, the CADs represents also an approach that is extensible, once dimensions can be added or better refined and flexible, once dimensions can be removed or added enabling focused analyses.

Clearly, the interpretation of scales for the dimensions chosen can vary according to the context. However, it is a general interpretation that is provided by the CADs. Once the concepts cannot (in principle) be numerically evaluated and compared, it is their semantic meanings and interpretations that must be taken into account. The proportions are also empirically associated with the dimensions, because no formal experiments were conducted so far to identify real metrics that could be then associated with each dimension and its respective granularity levels. For each case of application of the CADs, its use must be defined and discussed. It is worth to note that all CADs dimensions, although comprised in the same representation, are still independent, and as such the concentric circles while aid the comparison of different granularity levels, do not necessarily represent the same coverage level between two different dimensions.

The central circle of the CADs, colored in red, represents the absence of adaptation concerns, for example when no adaptation process is performed an application can be classified as designed regarding its autonomy level. For each subsequent circle an additional coverage level of adaptation can be considered, and more external levels represent a higher coverage regarding one specific dimension for adaptation. So, for in-stance, an application able to adapt regarding many modalities (multi) can be classified as having a higher coverage level of adaptation regarding the modality dimension if compared with another application that performs adaptation within the same modality type (intra).

It is worth to note though, that a higher coverage level of adaptation regarding one specific dimension does not immediately imply a higher level of usability or a better application for the end users. Implementing adaptation imposes many trade-offs (e.g., adapting an application may negatively affect its performance or accessibility level), and thus only by carefully planning and performing evaluation sessions, the actual benefits of adaptation for end users can be known.

The current version of the CADs results from the iteration between continuous evaluation and improvements, and thus it maintains the advantages of its previous version and discards potential issues that could lead to misunderstandings. This section explains the characteristics of the CADs, highlights its advantages and discusses its weaknesses.

As mentioned above the CADs diagram is extensible and flexible, therefore dimensions can be removed, inserted, or refined. Clearly, for analyses that require more focus, a specific set among the CADs dimensions can be selected. On the other hand, for broader analysis it is also possible to include and consider further dimensions and granularity levels.

1.3.4 Run-time components

1.3.4.1 Runtime UI generation engine (RUIGE)

The Runtime UI Generation Engine (RUIGE) module is intended to generate Service Front Ends (SFEs) that have been previously defined by means of ASFE-DL descriptions and to adapt the resulting SFEs in accordance the Adaptation Engine decisions. RUIGE has been defined as a set of sub-modules, in such a way that each of them is specialized in the generation of a specific kind of target platforms and modalities.

Each RUIGE sub-module is composed of three technological components in charge of (a) the transformation process from ASFE-DL to the corresponding CUI (b) the generation operations from CUI to FUI and (c) the runtime support for context-aware adaptation. The status of RUIGE is therefore defined by describing the status of each sub-module.

MMW-4S sub-module:

- MMW-4S transformer has been implemented as an XSL transformation sheet (XSLT) due to the fact that both the input language (ASFE-DL) and the output languages (IDEAL2 and SCXML) are XML-based. The modifications carried out over the MMW-4S Transformer were intended to provide support for the last specification of ASFE-DL, mainly the back class and the role attribute inclusion.
- Regarding the generation process, minor changes have been performed in order to improve the organization of the generation of the set of JSPs that guide the execution of the HTML5 Rendering Engine at runtime.
- In what regards to the generation process, more IDEAL2 User Interface components have been considered, implementing its rendering by means of the jQueryMobile library. Moreover, the integration with the Adaptation Engine has been improved.

MARIA sub-module:

- The MARIA transformer is implemented through an XSLT stylesheet. In the last year, we have particularly focused on the transformation for the MARIA Multimodal concrete

description that will be the input for the generator module. This concrete language combines, in different ways, graphical and vocal interactions using CARE (Complementarity, Assignment, Redundancy, Equivalence) properties associated with the various user interface elements.

- The MARIA Multimodal Generator is also implemented with an XSLT stylesheet and produces JSP implementations combining graphical vocal interfaces in Web environments. For this purpose, an Android app (for mobile devices) able to interpret the HTML generated and which call Google APIs for Automatic Speech Recognition (ASR) and Text-To-Speech (TTS) synthesis were developed, while for desktop applications it is sufficient to use the last versions of the Chrome browser.
- The MARIA Runtime integrates the communication with the Adaptation Engine and the Context Manager. In multimodal adaptation, the action part of the rules coming from the Adaptation Engine changes the CARE values of the interface elements adding or removing the graphical or the vocal modalities. An adaptation script, inside the generated final user interface, receives the action part of the rule in JSON format and interprets the received CARE values by adding or removing CSS class (as it was aforementioned) of the interface elements in order to adapt the interface. The implementation of the MARIA Runtime has also been exploited in the SAP prototype for warehouse picking.

UsiXML sub-module:

- The last version of the UsiXML sub-module consists of: a graphical editor for editing adaptation rules according to an interpretable syntax and a rule engine that processes rules at run-time. Parts of the code of this rule engine have also been incorporated in Quill (W3C).
- The execution at runtime varies according to the technology used to generate the FUI. Specific approaches are applicable in each case: for the HTML document or the Java Swing version, the runtime must be respectively performed with the browser itself or the java compiler.

Avatar sub-module:

- The UI Transformer outputs the Serenoa Avatar CUI, which is based on the FML language proposed by SAIBA framework from the abstract descriptions of the avatar. However, the transformer does not fully implement the capabilities of FML. For instance, FML covers gesture-text synchronization and fine descriptions for the elements that conforms the gestures, but the SERENOA Avatar CUI only describes a generic gesture.
- The UI Generator incorporates the remaining elements of FML that have not been included in the aforementioned transformation process. In order to achieve this goal, the results of the transformer are converted to a BML-like language that will be used for the different target platforms.
- The UI Generator provides different execution profiles depending on the performance conditions. For example, an ActiveX component in case of the runtime environment is an Internet Explorer browser or a sequence of images or videos in case of a mobile device platform. The UI Runtime is in charge of interpreting the FUI description of avatar actions and actually rendering an avatar that acts in accordance to them.

Leonardi:

- The LEONARDI Generator is implemented and already operational, as illustrated by the current status of the eCommerce scenario. It uses as an input a LEONARDI model, using the LEONARDI-DL as syntax. This mechanism allows benefiting as much as possible of the LEONARDI environment for Serenoa purposes. It relies on the LEONARDI execution engine to interpret, on the fly, the application model. The LEONARDI engine is implemented in the Java technology. It is used as a service by fat clients (Java applications), typically used as desktop applications, and deployed as a Servlet for web based applications (including mobile applications).

- The LEONARDI runtime takes advantage of the LEONARDI technology and integrates the communication with the Serenoa Adaptation Engine and the Context Manager. For a specific user and context, the application queries the Adaptation Engine for the best adaptation rule based on the user's profile and context providing user ID and service ID), and then effectively adapts the UI for the specific platform based on the response returned by the Adaptation Engine. The LEONARDI Runtime is completed by specific modules for mobility, as an "iOS Player" and an "Android Player".

1.3.5.2 Context Manager

A Context of Use Runtime Infrastructure has been developed to support the Context of Use management in the Serenoa framework. This layer is composed of several modules. The main one is the Context Management Core, which hosts several sub-modules such as a data store that keeps information and an interface for interaction with other external modules/components. Examples of such external components include both those that provide context information (e.g. context delegates of user devices, DDR...), and components that make use of context information (e.g. CARFO, Adaptation Engine...).

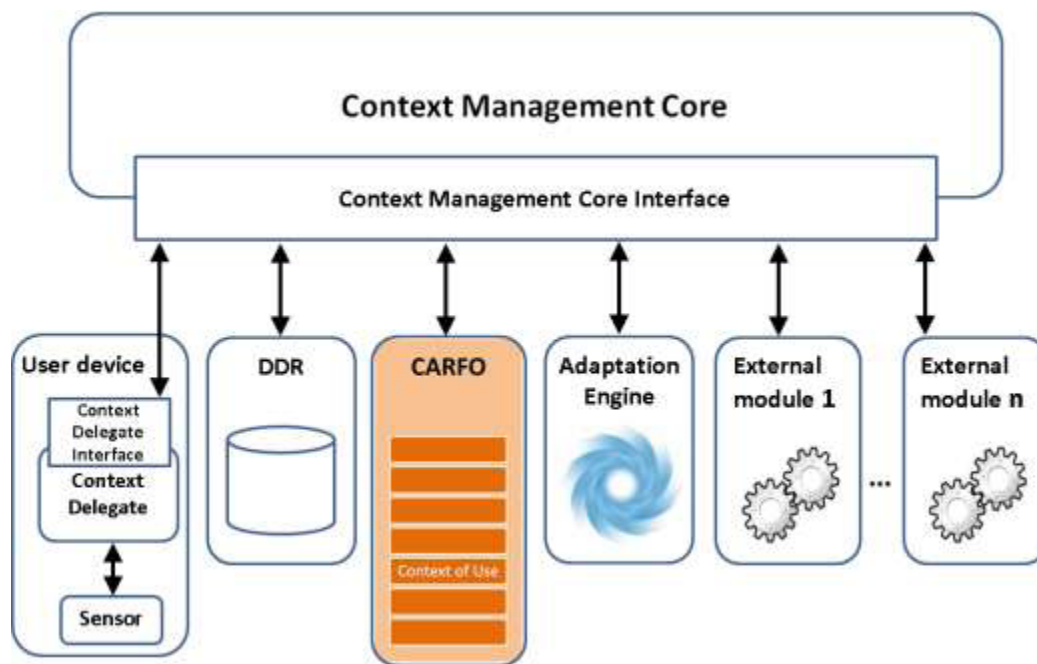


Figure 8. Architecture overview of the context management support

In particular, the Context of Use management support consists of several modules for gathering context information, saving and exposing it to other modules/components of the framework (e.g., Adaptation Engine). During the project we have refined its novel functionalities and data modeling strategies.

The Context Manager has a client/server architecture (see Figure 8). The context events are detected by Context Delegates (small applications installed on the devices), which monitor environment parameters and send updates to the Context Manager.

Various modules of the SERENOA framework have been integrated with the Context Manager as well. Such modules include context delegates deployable on user devices, providing context information to the Context Manager, as well as major components of the Serenoa framework. Examples of context delegates provide information regarding environment noise and light, Bluetooth beacons availability, physiological parameters, device battery. An example of components of the SERENOA framework that has been integrated with the context manager is the

Adaptation Engine that subscribes to particular events in order to get asynchronous notifications about relevant context changes.

An implementation of the context of use management support is available and has already been exploited by some project partners (e.g. remotely accessed and/or integrated in prototypes).

1.3.4.2 Adaptation Engine

By using an authoring environment at design time, it is possible to obtain both the logical specification of the interactive application and the specification of the rules to be used for adaptation (“Adaptation Rules”) which are both passed as input to the Adaptation Engine. The latter determines the optimal adaptation for the interactive system at hand in the current context of use, and based on the specified adaptation rules. To achieve this goal, the Adaptation Engine receives from the Context Manager information about the current context (e.g. events detected), and checks whether some adaptation rules could be triggered, by analyzing if the event/condition parts of some rules are satisfied. In the positive case, the corresponding action part of the selected rule is sent to the module in charge to perform the associated modifications.

1.3.5 Applications prototypes

The [Warehouse-Management Scenario](#) for the Intelligent-Picking prototype (see Figure 9) is aimed at providing a seamless context (environment and task) adaptation experience to users in one of the partner's Living Lab facilities in the field of Future Retail Concepts (FRC).

In order to ensure a just-in-time and just-in-sequence deployment of relevant components to the respective assembly stations, many companies define a proceeding consignment step. Here workers run off storage racks in a warehouse intending to pick the necessary parts. The workers of the succeeding assembly step strongly rely on a correct consignment. An uncompleted consignment will lead to a downtime of the production line, which is translated into losses for the company. In a similar manner, inaccurate pickings will decrease the quality and increase workers' frustration. Therefore, the picking process makes high quality and time demands on the workers. Especially the fact that warehouses employ unskilled workers to relieve the workload during peak times makes consignment a critical task and bottleneck in the overall process. The problem is that workers are often unfamiliar with warehouse settings.

This scenario motivates how proactive applications can provide unobtrusive and adequate help (e.g. missing parts, location of necessary parts, etc.) when the user needs help. Thereby, the service time can be reduced while increasing the quality of service.

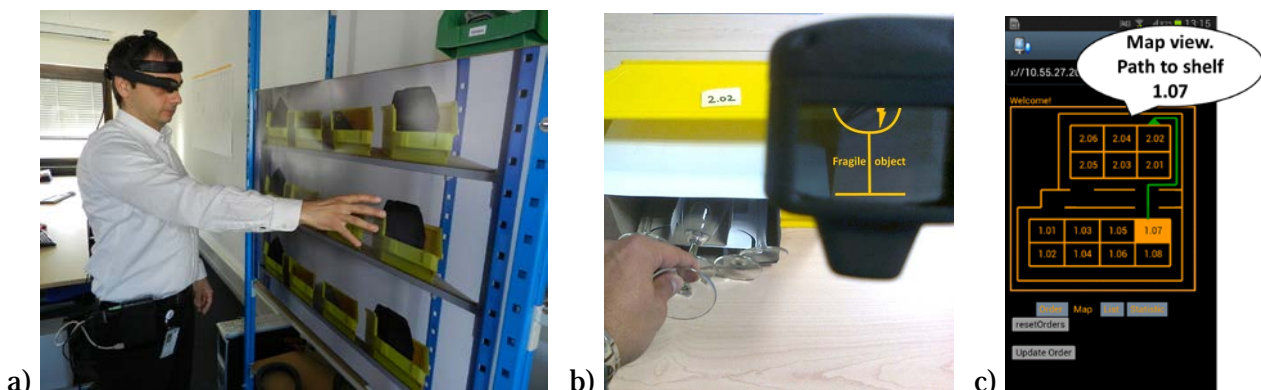


Figure 9. Warehouse-Management scenario: a) The Intelligent-Picking prototype allows hands-free picking through smart glasses and vocal interaction. b) Adaptation of the UI when picking a fragile object. c) Heads-up navigation to the shelves with vocal output.

The Intelligent-Picking prototype (IPP) demonstrates the enhanced user experience obtained through adaptive user interfaces. Compared to the traditional interaction, in this prototype the UI is fully audio-based: the users are hands-free when they interact with it. As the scenario requires hands-free interaction the prototype is based on audio in- and output. However when the interaction only relies on the vocal modality it bears the problem of imperfect speech recognition and the burden of memorizing the audio commands. In conjunction with the environmental conditions, e.g. noise, the audio interface alone does not always lead to a good user experience. The adaptability of the UI opens the possibility to work with multiple interaction modalities i.e. vocal, visual and tactile. The UI adapts by reacting to changes in the context (e.g. noise level of the environment) and choosing suitable combination of modalities for interaction. The self-adaptive features enable users to finish the picking task more efficiently and effectively.

The IPP is integrated in the Serenoa framework and related to other supporting modules of the Serenoa framework. From the application scenario a set of adaptation use-cases can be defined using the patterns and format described in Reference Models. From these use-cases we are able to describe the relevant adaptation rules in terms of events, condition, and actions according to the Serenoa language for adaptation rules using the Authoring tool. These rules are uploaded to an Adaptation Engine and triggered through the events from the Context Manager. The adaptive features of the UI can be realized through the use of model-based descriptions of interactive applications. At design time the various initial versions of the applications are developed in the abstract model according to the Serenoa language. The generator based on the MARIA language (RUIGE) produces the final IPP interface. The IPP communicates with the Adaptation Engine to receive adaptation actions triggered by the Context Manager.

The **E-Health Scenario** is shown based on the adaptation of an existing e-health prototype previously developed by Telefónica to ensure the maximization of the exploitation capabilities (see Figure 10).

The SARA project is intended to provide a user interface for chronic disease patients self-monitoring in the form of a (Windows based) tablet PC. The project wants to evolve to provide multi-device support (Android tablet devices, smartphones, etc.) and a virtual assistant in order to engage patients in the usage of the application. This project is currently in a pre-market phase, after successful field tests using real patients from the Andalusian health system. The inclusion of ECAs (Embodied Conversational Agents) technology for guiding the navigation through all this personal information is a key factor for improving usability.

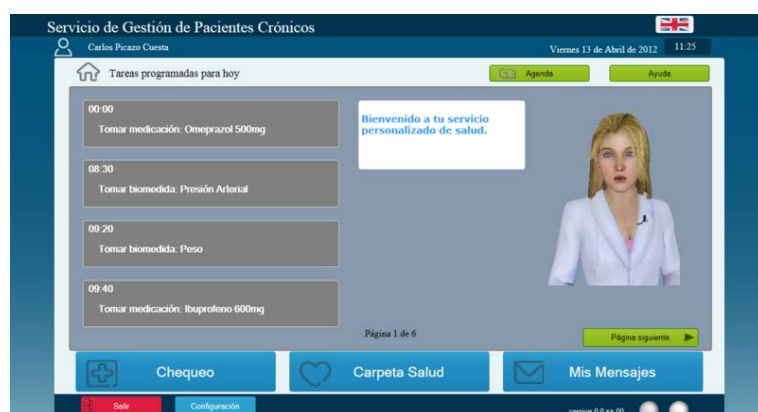


Figure 10. E-health scenario

Trying to adapt the E-Health prototype to mobile devices, a mobile-device application has been developed using the Phonegap/Cordova framework. Android has been selected as the preferred target platform for mobile devices, since Android devices are currently the most sold platform worldwide.

Different features of the desktop application have been ported to the mobile one. The language and avatar preferences of the application are retrieved from the Context Manager. If the user preferences state that the avatar must be shown, we place a video of the avatar instead of the

ActiveX component used in the desktop version of the application, because the mentioned component is not suitable to work with Android applications.

Figure 11 shows the main screen of the SARA project adapted to mobile and tablet devices.

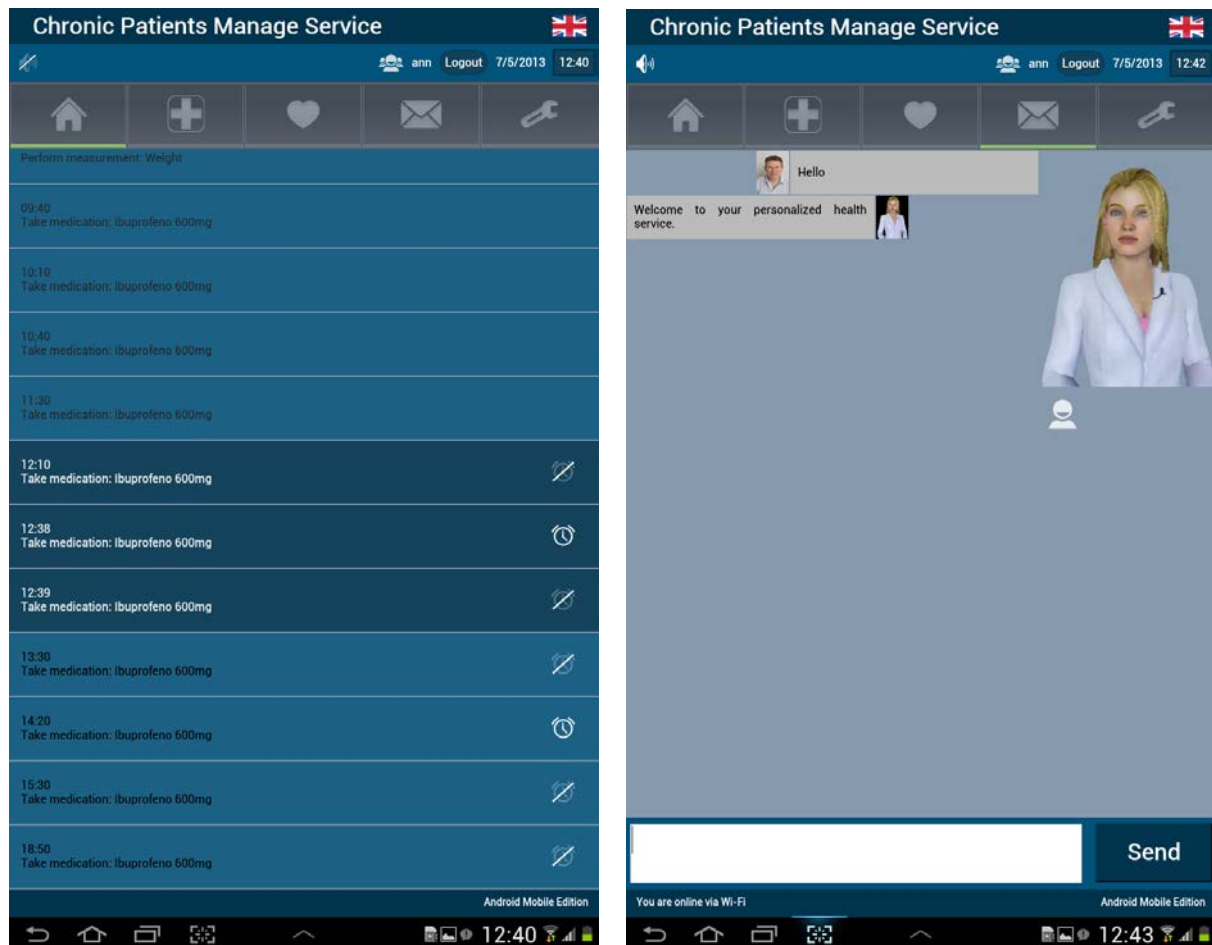


Figure 11. SARA E-health scenario adapted to Android-based mobile devices

Another example of adaptation is the one related with the noise level. If the noise level surrounding the user is low enough, then the text will be read by the avatar. When the noise level is higher, it is preferable not to read the text, because the user may not be able to hear it. So, the avatar is muted. A context delegate application continuously senses the noise level detected by the device microphone and sends the value to the Context Manager. When some relevant event occurs (e.g. when the noise level exceeds a predefined threshold), the Context Manager fires an asynchronous notification to the Serenoa Framework and the RUIGE adapts the text-to-speech functionality.

In addition, the application has a notification service to remind the user which actions should be performed as recommended by the doctor, such as taking pills and visiting the doctor. As the application makes use of Android notification service, all notifications will be available although the application was not open (see Figure 12).

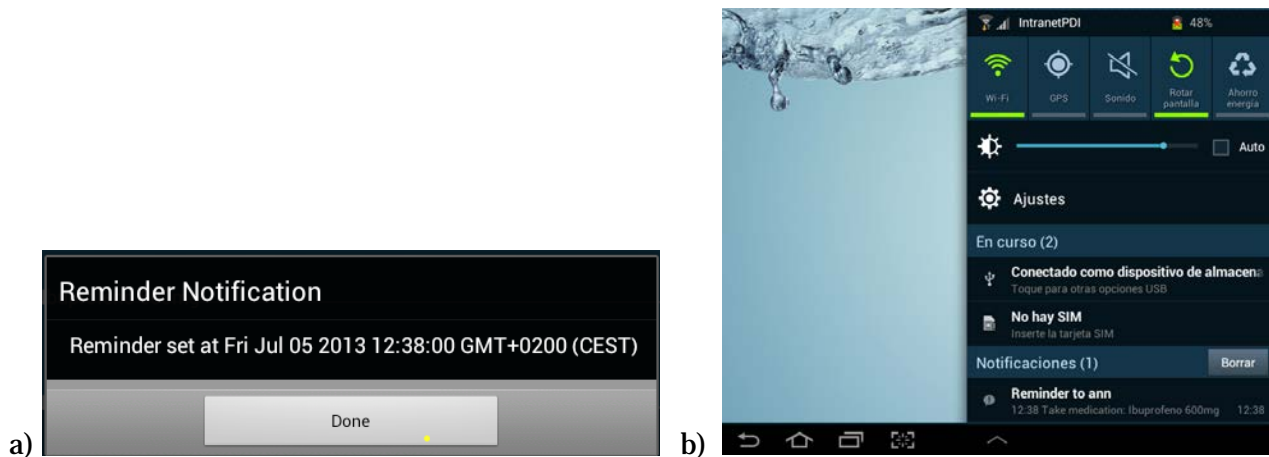


Figure 12. a) Notifications details in a notification. b) Notifications can be performed even when the application is closed

With the [E-commerce prototype](#), we aim at illustrating how a business application can make use of the Serenoa platform to provide context sensitive features to improve the end-user experience. Our intent is to show how, in a typical E-Commerce example; different online end-users can take advantage of adaptive SFEs while connecting to both a front-end application (for on-line shoppers) and a back-end application (for employees). Typical user roles involved in the scenario include online shoppers and employees, acting either as supervisors or customer representatives in charge of following-up with online orders.

Based on their roles, such users can access different features in the E-commerce prototype and their UIs are capable of adapting based on different factors such as language, color-blindness, type of device (either a desktop computer, a tablet or a smartphone), or even gesture interaction (by using a head-tracking device). In Figure 13, some screenshots show the front office prototype when connecting from a desktop, an iPhone and an Android tablet.

The E-Commerce prototype reached its objectives in terms of illustrating how the Serenoa platform can be used for adaptation of a business oriented application; it was initially based on W4's LEONARDI technology and then integrated progressively with the Serenoa framework, as the different software modules became available and matured. In its final state, it is based on the rules repository (rules expressed using the ASFEDL), it is connected to the adaptation engine and the context manager, and it makes use, at runtime, of the LEONARDI RUIGE module.

The result of the development for the prototype was reused at W4 as the basis for a significant new feature in W4's offering, the "W4 Player". This consists of a software program, now available in both *Apple's App Store* and *Android Google Play*, which can be installed on iOS and Android devices to connect to any W4 business application deployed on a web server using a native application.

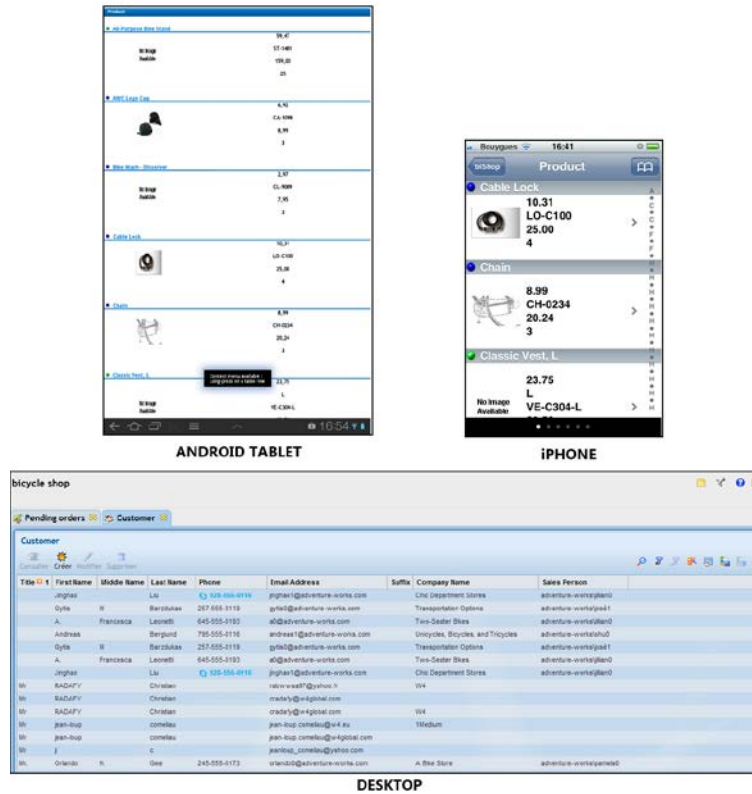


Figure 13. E-commerce prototype screenshots

1.3.6 Main S&T Results

SERENOA S/T results can be summarized as follows:

1.3.6.1 SERENOA Architecture

The final version of the Serenoa architecture was shown in Figure 1. This architecture is based on a modular framework. It means it is composed of different sub-modules, each one in charge of a particular functionality. All of them are communicated in order to accomplish the generation of context-aware SFEs.

One of the components involved in this architecture is the Runtime User Interface Generator Engine (RUIGE). This module is responsible for generating and executing applications for different platforms from an abstract description of the User Interface (ASFE-DL). The generated applications are adapted to the delivery context. RUIGE needs to interoperate with other modules of the Serenoa platform such as the Adaptation Engine to carry out such adaptations. RUIGE is also split in diverse sub-modules allowing the generation, deployment and execution of an application for different interaction modes and target platforms as, for example, a runtime in charge of generating mobile applications, voice applications, avatar engines, etc.

Different stages are required to generate the final application in each of the RUIGE sub-modules: the **transformer** to convert from the abstract description (ASFE-DL) to a special representation format that could be interpreted by each one of the generators; the **generator** takes the concrete language obtained in the previous phase and generates the executable/interpretable code and, finally, the **runtime** which provides the runtime support for the application execution itself (it could be carried out by a virtual machine, a web browser, an operating system executing machine, etc.). However, each RUIGE sub-module might include any other components.

Five RUIGE sub-modules have been developed:

- **MMW-4S** is a RUIGE sub-module able to generate mobile Web UIs for different browsers and devices.
- The **MARIA** RUIGE sub-module is able to support the adaptive Service Front Ends through Graphical, Vocal and Multimodal interfaces. The final output of can use different web technologies depending on the interaction modality: JSP + HTML5 for graphical interfaces, VoiceXML for vocal interfaces, and HTML, JavaScript plus Google support for Web Speech API.
- The **UsiXML** RUIGE sub-module is responsible for the generation, deployment and execution of applications, based on a specification in ASFE-DL of the AUI. To accommodate different contexts, UsiXML provides generation of UIs in HTML and Java Swing.
- The **AVATAR** sub-module is in charge of generating adaptation for Service Front Ends based on a three-dimensional, human-like avatar.
- The **LEONARDI** RUIGE sub-module generates both typical B2B applications (the back-end system) and B2C applications (the front-end system). It is possible to visualize resulting applications from a desktop web browser and from various types of mobile devices such as iOs and some Android tablets and smartphones.

1.3.6.2 Languages

One important output has been the languages for specifying logical descriptions of interactive applications (ASFE-DL) and adaptation rules (AAL-DL). The languages support the development of multi-device user interfaces and their ability to adapt to the changing context of use.

1.3.6.3 CARFO Ontology

The CARFO ontology, an OWL2 ontology in the shape of a Context of Use module for SFE adaptations, is available for both human and machine consumption at <http://purl.org/carfo>.

The ontology population process has been achieved in an automatic way using data sources such as caniuse.com, CTIC Device Description Repository (DDR) and Nokia Developer site. Also, a successful proof of concept of the generated knowledge visualization has been carried out, using RDF Data Cube compliant tools.

All the code used to achieve the ontology population is available as Open Source at <https://code.google.com/p/serenoa>.

All the generated RDF triples are available from an SPARQL endpoint at <http://data.ctic.es/sparql> (at the named graph <http://purl.org/carfo/example>), showing the data results of the work done.

1.3.6.4 Authoring Tools

In the framework of “Authoring Environment and Development Tools”, we have produced two types of authoring tools/environments, namely Eclipse-based plugin and the HTML5 browser-based application, which would make it easier and more efficient for the developers to build adaptation rules and service UIs descriptions using the AAL-DL and ASFE-DL languages, respectively. These descriptions are then sent to the service repository hosted inside the Adaptation Engine, where they are accessible via an REST interface. On the other hand, a remote service repository is also integrated (via REST interface) with the authoring tools to support the CRUD (Create, Read, Update, and Delete) operations for these descriptions.

1.3.6.5 Prototypes

An Intelligent-Picking prototype (IPP) which demonstrates the enhanced user experience obtained through adaptive user interfaces. The prototype works with multiple interaction modalities i.e. vocal, visual and tactile. The UI adapts by reacting to changes in the context (e.g. noise level of the environment) and choosing suitable combination of modalities for interaction. The self-adaptive features enable users to finish the picking task more efficiently and effectively. The IPP is

integrated in the Serenoa framework and related to other supporting modules of the Serenoa framework.

The eHealth prototype shows different adaptation capabilities, to the noise level, to the user and to the mobile device and it is linked to a strategy area of Telefónica R&D, so exploitation opportunities are strong.

The E-Commerce prototype reached its objectives in terms of illustrating how the Serenoa platform can be used for adaptation of a business oriented application; it was initially based on W4's LEONARDI technology and then integrated progressively with the Serenoa framework, as the different software modules became available and matured

1.4 Potential impact & main dissemination activities

1.4.1 Added value

In a general way, the Serenoa project may contribute to the following aspects:

- **Faster time-to-market:** given the support provided by the Serenoa technologies (i.e. authoring tools, adaptation languages and models) the implementation of adaptive and adaptable applications will be easier and more efficient.
- **Reduction of development effort:** instead of spending a lot of resources looking for knowledge about context-aware adaptation, stakeholders can rely in a centralized information source and a dedicated development platform, provided respectively by the theoretical framework, and by the technological framework (i.e. tools, languages, etc.).
- **Promote re-use:** by means of standards and a consistent terminology, the applications can be implemented in a more flexible manner, i.e. with the integration of components and with the extension and updates of existing applications.
- **Efficient solutions:** Serenoa is filling a gap in context-aware adaptation systems by allowing non-expert developers to develop efficient solutions. This is achieved, in particular, by adopting a model-based approach.
- **Increased agility:** because context is often dynamic by nature, adaptation needs to be tuned little by little (and sometimes continuously), based on the feedback provided by the different stakeholders, starting with the end-user. Therefore, the capability for systems to change its behavior rapidly and to take into account new adaptation rules, thus fitting new contexts, is essential. Serenoa's adaptation engine adds substantial agility to the development process of context-aware systems, which can provide great benefits.
- **Joining efforts:** By analyzing the technological landscape and the current status of the market, Serenoa identifies actual users' needs and orient its efforts in order to progress simultaneously in both scientific and industrial domains. Besides this, Serenoa tries to establish a formal link between the research community and the industry (for instance by means of standardization and dissemination actions).

1.4.2 Impact

As a part of the individual exploitation strategies the consortium members demonstrate the Serenoa 'Applications prototypes' and Development Tools to product groups, customers and partners. This aims at encouraging these industrial stakeholders for an adoption of the Serenoa technologies to enhance their existing products or to create new ones.

From the point of view of the academic community, several branches of knowledge are involved with research of context-aware adaptation. As the closest related ones, we can highlight: Human-Computer Interaction, Software Engineering and Architecture, Distributed Systems and Ubiquitous Computing. In the context of Serenoa project, the scientific field may benefit from: concrete requirements for designing and implementing adaptive systems, authoring tools and languages that support the creation of adaptive applications, theoretical frameworks that provide a catalogue of information to support the research in the field, evaluation criteria and possible architectural approaches.

Furthermore, releasing some Serenoa components as open source facilitates the adoption of the results by other members of the community, even any other projects, communities, organizations or anyone interested in this focused area. This issue implies that any of them could take the results and evolve them into something more complete, allowing to increase the impact of the project outside the community of the project. Specifically, Serenoa is expected to produce three types of results: theoretical frameworks, languages, and application prototypes.

- **Theoretical frameworks:** the reference models (CARF and CADS) have been made available for the public by means of written documents (as deliverables and scientific papers). They also include a detailed description about their methodology, creation, and application available online by means of a web page. The Ontology has been developed based on this theoretical framework.
- **Languages:** results on this field have been submitted to W3C for standardization. The W3C working group on Model-Based User Interfaces (MBUI) have been very active and various Serenoa partners have been involved in this group.
- **Application prototypes:** the applications, which illustrate the Serenoa framework, lay on various software modules: context manager, adaptation engine, runtime engine and the rest of components. In general, the software prototypes have been made publicly available. Several policies for making them public have been envisioned: open source, public executable code, videos, etc. The way how they have been made public depends on the organization that is developing them and the type of prototype.

In order to accomplish these results and to arouse interest in the scientific and industrial community, an action planning was envisaged. Firstly, on August-September 2012, the final version of the Serenoa's architecture was delivered and the second release of the modules was available for the evaluation. From then to the end of the project (September 2013) the development efforts have been mainly focused on the final release of the Serenoa's components and tools, the exploitation of them in the Serenoa use cases and the final evaluation.

1.4.3 Serenoa beneficiaries

Serenoa is a framework that simplifies and accelerates the development of context-sensitive applications. This implies the reduction of the time-effort of the development process and its corresponding maintenance costs. The value of the framework is based on a solid model-based architecture, a good methodology and advanced tools to automatically generate applications. Different user roles are susceptible to take advantage of the aforementioned features:

- **Developer:** both final applications developers and those who are interested in adaptive technologies.
- **Researcher:** scientific community interested in context-aware adaptation and related topics.
- **End-User:** indirect beneficiary who gets faster results because of the time-to-market reduction.
- **Executive Director:** professional who may be interested in the creation of new business opportunities.

1.4.4 Industrial Advisory Board



Figure 14. Industrial Advisory Board Meeting 2013

1.4.4.1 Overview

During the project, Industrial Advisory Board (IAB) meetings were held at the end of each one of the three project periods, as planned in the initial description of work.

The intent of the IAB is to narrow the gap between the R&D efforts that occur during the project on the one hand, and the business community's expectation in terms of adaptation features, on the other hand. Another stated objective is also to come up with suggestions from industrial actors to improve the efficiency of the exploitation actions in regards to the project results.

1.4.4.2 Participants

Each one of the organized IAB face-to-face meetings involved, in addition to the project representatives, business stakeholders who were invited at each meeting to provide relevant feedback. To make sure that we kept in mind the "big picture", attendees included representatives of the industrial project stakeholders (SAP, TID, W4), but also external people.

1.4.4.3 Format

Each IAB followed the same format. Some preparation work was first suggested to attendees (three weeks ahead of the meeting), chosen to be in relation with the current context of the project. These topics were:

- Year 1: general market drivers that can impact or influence context-aware SFEs and business use cases that illustrate this field in your business landscape.
- Year 2: discussion around the collaborative white paper and thoughts about relevant personas matching the described ones in the white paper.
- Year 3: relevant actions for exploitation plans.

Then, the meeting was organized as following:

- It began with a presentation about the results obtained on the project during the previous period and, when possible, with demos.
- Then, a section was dedicated to discussing the topic chosen as preparation work.
- Then, there was an open discussion section where feedback was gathered in reaction to the first two slots.
- A conclusion was made, highlighting the key points raised during the meeting.

While the IAB meetings at the end of year 1 and year 2 were mostly focused on market drivers and use cases the last meeting was completely dedicated to exploitation plans.

1.4.4.4 Key points

Below is a list of the main key points coming out from the three meetings, as reported in the associated deliverables:

- There are huge expectations from the business community in terms of adaptive SFEs in many, diverse fields.
- The initial goals of the project were very ambitious and restricting its scope and identifying the personas was useful in order to achieve results.
- Mobility, BYOD and social medias are the main ongoing drivers that keep influencing the Serenoa market scope.
- The exploitation plans, as exposed to the participants in the third meeting, is relevant.

1.4.5 Main dissemination Activities

During the 3 years of the project, the SERENOA partners were highly active in disseminating the project's results in high-quality conferences and journals, liaising with related research projects, demonstrating the SERENOA prototype and contributing to relevant standardization bodies. 41 scientific papers have been prepared, submitted, accepted for publication, and presented in conferences, besides this 33 dissemination events have been conducted, including industrial workshops attended, keynote talks presented, standardization meetings and fairs. The project's main dissemination activities are organized below in 3 categories: academic, industrial dissemination, and standardization.

1.4.5.1 Academic Activities

The academic dissemination has been characterized by:

- Scientific papers that have been prepared, submitted, published, presented and discussed during conferences and workshops (as AVI, CHI, EICS and Interact);
- Tutorials and short courses that have been planned, organized and given during scientific conferences (as Interact and CHI);
- Journal papers that have been written, submitted and published (as Psychology and RRIOC);
- Keynote speeches presented during scientific conferences and events (as Quindio and Interaccion);
- Workshops that have been proposed, planned, organized and conducted (as CASFE and DUI).

The academic dissemination targeted at the scientific audience and researchers, however the related events, as conferences and workshops also involved practitioners and participants from the industrial fields.

1.4.5.2 Scientific Conferences

The dissemination activities covered a set of different events, including the following scientific conferences:

- ACM Conference on Human Aspects in Computing Systems (CHI)
- IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)
- IFIP Conference on Human-Computer Interaction (Interact)
- ACM Symposium on Engineering Interactive Computing Systems (EICS)
- International Conference on Web Engineering (ICWE)
- World Wide Web Conference (WWW)
- MobileHCI
- International Working Conference on Advanced User Interfaces (AVI)
- International Joint Conference on Ambient Intelligence (AmI)
- International Conference on Research Challenges in Information Science (RCIS)
- International Conference on Language Resources and Evaluation (LREC)

1.4.5.3 Industrial Dissemination

The industrial dissemination of the project is characterized by attending events and fairs. TID, W4, SAP and W3C have been engaged in such events, disseminating the project by means of printed materials (as posters and newsletters), and oral presentations. These events enabled partners to exchange experiences with practitioners with similar interests and in related domains.

The industrial activities for dissemination of Serenoa also covered a set of different events, including the following ones:

- SAP TechEd
- SAP Handelssymposium in Frankfurt
- Telefónica I+D "Research Fair"
- Mobile World Congress
- Mobile Forum
- Model Driven Day in Paris
- Documation
- Solutions DEMAT'
- Mobility for Business
- FIA Dublin



Figure 15. Mobile World Congress 2013

Further details about the industrial and scientific efforts on disseminations are reported in the public deliverable D7.1.3.

1.4.5.4 Standardization

The standardization actions of the project have been concentrated in the Working Group of W3C. The W3C Model-Based User Interfaces Working Group was launched on 17 October 2011 and is chartered until the 13 November 2013. Work is proceeding with a mix of regular teleconferences, the mailing list and wiki, and face to face meetings. The final goal is to achieve a standard specification for the Task Model and the Abstract User Interface. To do so the partners mainly involved in such activity (W3C, CTIC, UCL and ISTI) have been actively participating in the group, regularly attending the weekly call meetings, collaborating in preparing the documents and models, and participating in periodic face to face meetings.

Four meetings were held for now:

- the first face to face was hosted by DFKI in Kaiserslautern, Germany on 9-10 February 2012,
- the second on 14-15 June in Pisa, hosted by ISTI-CNR,
- the third on 29-30 October 2012 as part of the W3C Technical Plenary in Lyon,
- the fourth on 10-11 July 2013 in Grasbrunn, near Munich, hosted by Red Hat

Further details about the efforts on standardization are reported in the public deliverable D6.2.2.

1.5 Project public website and contact details

All relevant work, including deliverables and published papers are available through the SERENOA website accessible at <http://www.serenoa-fp7.eu>

The screenshot shows the SERENOA project website. The header includes the Serenoa logo and the tagline "Multidimensional context-aware adaptation of Service Front-ends". Below the header is a navigation menu with links for HOME, OVERVIEW, PARTNERS, DELIVERABLES, DISSEMINATION, GLOSSARY, LINKS, and CONTACT. The main content area features a "Welcome to the SERENOA PROJECT" section, followed by a paragraph explaining the project's funding and goals. It mentions that the project is funded by the European Commission's Seventh Framework Programme under grant agreement n° 258030 (FP7-ICT-2009-5. Internet of Services, Software and Virtualization – STREP). The text describes the project's aim to develop a novel, open platform for enabling the creation of context-sensitive service front-ends (SFEs). It also lists partners, including Telefonica, and mentions a "White Paper" available for download.

Figure 16. Serenoa website

Contact Information for each project partner is provided in the following table:

| Partner | Contact | Address | e-mail | Phone |
|-------------------|-------------------|--|--|------------------|
| TID (Coordinator) | Javier Caminero | Ronda de la Comunicación s/n, 28050 Madrid, Spain | fjcg@tid.es | +34 914832671 |
| UCL | Jean Vanderdonckt | Place des Doyens, 1 – 1348 Louvain-la-Neuve, Belgium | Jean.vanderdonckt@uclouvain.be | +32 010 47 85 25 |
| ISTI | Fabio Paternò | Via G. Moruzzi 1, 56124 Pisa, Italy | fabio.paterno@isti.cnr.it | +39 348 396 2624 |
| SAP | Joerg Rett | Bleichstr. 8, 64283 Darmstadt, Germany | joerg.rett@sap.com | +49 6227 768879 |
| W3C | Dave Raggett | W3C/ERCIM, 2004 route des Lucioles, Sophia Antipolis, 06410 Biot, France | dsr@w3.org | +44 1225 867 351 |
| W4 | Jean-Loup | W4 S.A. Parc d'affaires | jean- | +33 15670 53 80 |

| | | | | |
|------|--------------------------|---|--|--------------|
| | Comeliau | SILIC, 46 rue de Monthléry, BP 70178, 94563 Rungis FRANCE | loup.comeliau@w4.eu | |
| CTIC | Javier Rodríguez Escolar | C/Ada Byron 39 33203 Gijón | javier.rodriguez@fundacionctic.org | +34984291212 |

2 Use and dissemination of foreground

2.1 Section A

| TEMPLATE A1: LIST OF SCIENTIFIC (PEER REVIEWED) PUBLICATIONS, STARTING WITH THE MOST IMPORTANT ONES | | | | | | | | | | |
|---|---|-------------|---|---------------------------|-----------|----------------------|---------------------|----------------|---|--|
| NO. | Title | Main author | Title of the periodical or the series | Number, date or frequency | Publisher | Place of publication | Year of publication | Relevant pages | Permanent identifiers ³ (if available) | Is/Will open access ⁴ provided to this publication? |
| 1 | Model-based User Interface Adaptation in Multi-Device Environments | ISTI | Workshop on Dynamic Accessibility: Accommodating Differences in Ability and Situation, in conjunction with the ACM CHI conference | Annual (May 2011) | | Vancouver (Canada) | 2011 | | | |
| 2 | A SOA based Context-Aware Order Picking System for Warehouses using Laser Range | SAP | IEEE WoWMoM 2011 | June 20-24, 2011 | IEEE | Lucca (Italy) | 2011 | 8 | http://dx.doi.org/10.1109/WoWMoM.2011.5 | Via IEEE |

³ A permanent identifier should be a persistent link to the published version full text if open access or abstract if article is pay per view) or to the final manuscript accepted for publication (link to article in repository).

⁴ Open Access is defined as free of charge access for anyone via Internet. Please answer "yes" if the open access to the publication is already established and also if the embargo period for open access is not yet over but you intend to establish open access afterwards.

| | | | | | | | | | | |
|----|--|-----------|--|-----------------|--------------------|--------------------|------|----|---|----------------|
| | Finder and Wearable Computer | | | | | | | | 986185 | |
| 3 | Showing User Interface Adaptivity by Animated Transitions | UCL | ACM EICS | June 2011 | ACM DL | Pisa (Italy) | 2011 | 10 | http://doi.acm.org/10.1145/1996461.1996501 | Via ACM DL |
| 4 | Flippable User Interfaces for Internationalization | UCL | ACM EICS | June 2011 | ACM DL | Pisa (Italy) | 2011 | 5 | http://doi.acm.org/10.1145/1996461.1996524 | Via ACM DL |
| 5 | Engineering JavaScript State Persistence of Web Applications Migrating across Multiple Devices | ISTI | ACM EICS | June 2011 | LNCS – Springer | Pisa (Italy) | 2011 | 4 | http://doi.acm.org/10.1145/1996461.1996502 | Via Springer |
| 6 | Supporting Transformations Across User Interface Descriptions at Various Abstraction Levels | ISTI | IFIP INTERACT 2011 | Sep. 5-10, 2011 | LNCS - Springer | Lisbon (Portugal) | 2011 | 4 | http://doi.acm.org/10.1007/978-3-642-23768-3_94 | Via Springer |
| 7 | Adapting Desktop Web Pages for Vocal Browsing | ISTI | IFIP INTERACT 2011 | Sep. 5-10, 2011 | LNCS - Springer | Lisbon (Portugal) | 2011 | 8 | 10.1007/978-3-642-23765-2_42 | Via Springer |
| 8 | Model-Based Customizable Adaptation of Web Applications for Vocal Browsing | ISTI | ACM SIGDOC 2011 | Oct. 3-5, 2011 | ACM DL | Tirrenia, Italy | 2011 | 7 | http://doi.acm.org/10.1145/2038476.2038493 | Via ACM DL |
| 9 | Past, present, and future of model-based user interface development | ISTI, UCL | i-com Zeitschrift für interaktive und kooperative Medien | November, 2011 | Oldenbourg, Verlag | | 2011 | 9 | http://www.oldenbourg-link.com/doi/abs/10.1524/icom.2011.0026 | Via Oldenbourg |
| 10 | Automatic Reverse Engineering of Interactive Dynamic Web Applications to Support Adaptation across Platforms | ISTI | ACM Conference on Intelligent User Interfaces 2012 | February 2012 | ACML DL | Lisbon, Portugal | 2012 | 9 | 10.1145/2166966.2167004 | Via ACM DL |
| 11 | Making User Interface Adaptation in Multi-Device Environments Understandable to End Users | ISTI | CHI 2012 workshop: End-user interactions with | May 5-10, 2012, | - | Austin (USA) | 2012 | 7 | - | - |

| | | | | | | | | | | |
|----|--|------------------------------------|---|---------------------|--|-------------------|------|----|---|--------------|
| | | | intelligent and autonomous systems, | | | | | | | |
| 12 | The SERENOA Project: Multidimensional Context-Aware Adaptation of Service Front-Ends | TID, UCL, ISTI, SAP, W3C, W4, CTIC | Proceedings of the 8th LREC International Conference | May 2012 | Proceedings of the 8 Int. Conf. on Language Resources and Evaluation (LREC'12) | Istanbul, Turkey, | 2012 | 8 | http://www.lrec-conf.org/proceedings/lrec2012/pdf/516_Paper.pdf | |
| 13 | Animated Transitions between User Interface Views | UCL | AVI'2012 (Advanced Visual Interfaces) | May 22nd 2012 | ACM DL | Capri – Italy | 2012 | 7 | http://dx.doi.org/10.1145/2254556.2254623 | Via ACM DL |
| 14 | A Compositional Model for Gesture Definition | ISTI | HCSE 2012 - Human-Centred Software Engineering | October 29-31, 2012 | Springer | Toulouse (France) | 2012 | 19 | 10.1007/978-3-642-34347-6_3 | Via Springer |
| 15 | Multidimensional Context-Aware Adaptation of Service Front-Ends | TID | CASFE 2012 workshop collocated with AmI'12 International Joint Conference on Ambient Intelligence | Nov 13, 2012 | CEUR | Pisa (Italy) | 2012 | 5 | http://ceur-ws.org/Vol-970/paper1.pdf | Via CEUR |
| 16 | Opinion Mining Tools for the Analysis and Adaptation of Corporate SFEs | TID | CASFE 2012 workshop collocated with AmI'12 International Joint Conference on | Nov 13, 2012 | CEUR | Pisa (Italy) | 2012 | 4 | http://ceur-ws.org/Vol-970/paper2.pdf | Via CEUR |

| | | | | | | | | | | |
|----|--|------|---|--------------|------|--------------|------|---|---|----------|
| | | | Ambient Intelligence | | | | | | | |
| 17 | A Theoretical Framework for Specifying and Analyzing Context-Aware Adaptation | UCL | CASFE 2012 workshop collocated with Aml'12 International Joint Conference on Ambient Intelligence | Nov 13, 2012 | CEUR | Pisa (Italy) | 2012 | 4 | http://ceur-ws.org/Vol-970/paper4.pdf | Via CEUR |
| 18 | User-Centered Evaluation of an Adaptive User Interface in the Context of Warehouse Picking | SAP | CASFE 2012 workshop collocated with Aml'12 International Joint Conference on Ambient Intelligence | Nov 13, 2012 | CEUR | Pisa (Italy) | 2012 | 6 | http://ceur-ws.org/Vol-970/paper5.pdf | Via CEUR |
| 19 | A Manifest for Application Generators: Helping Developers with the Serenoa Framework | CTIC | CASFE 2012 workshop collocated with Aml'12 International Joint Conference on Ambient Intelligence | Nov 13, 2012 | CEUR | Pisa (Italy) | 2012 | 4 | http://ceur-ws.org/Vol-970/paper7.pdf | Via CEUR |
| 20 | How to Exploit Abstract User Interfaces in MARIA | ISTI | CASFE 2012 workshop collocated with Aml'12 International Joint Conference on Ambient Intelligence | Nov 13, 2012 | CEUR | Pisa (Italy) | 2012 | 5 | http://ceur-ws.org/Vol-970/paper8.pdf | Via CEUR |
| 21 | Machine Learning in the Support of Context-Aware Adaptation | UCL | CASFE 2012 workshop collocated with Aml'12 International Joint Conference on Ambient Intelligence | Nov 13, 2012 | CEUR | Pisa (Italy) | 2012 | 4 | http://ceur-ws.org/Vol-970/paper9.pdf | Via CEUR |
| 22 | A Set of Languages for Context- | ISTI | CASFE 2012 workshop | Nov 13, | CEUR | Pisa | 2012 | 8 | http://ceur-ws.org/Vol- | Via |

| | | | | | | | | | | |
|----|--|-----------|---|----------------------|-----------------|-------------------------|------|----|---|--------------|
| | Aware Adaptation | | collocated with Aml'12 International Joint Conference on Ambient Intelligence | 2012 | | (Italy) | | | 970/paper10.pdf | CEUR |
| 23 | Adaptive User Interfaces for Smart Environments with the Support of Model-based Language | SAP, ISTI | Aml'12 International Joint Conference on Ambient Intelligence | Nov 13-15, 2012 | LNCS - Springer | Pisa (Italy) | 2012 | 16 | 10.1007/978-3-642-34898-3_3 | Via Springer |
| 24 | The Design of Web Sites Adaptable to Emotion-related Aspects | ISTI | PsychNology Journal, 2012, Volume 10, Number 1, | 2012 | PsychNology | | 2012 | 15 | | |
| 25 | Adaptive Mobile Multimodal User Interfaces for Dynamic Accessibility | ISTI | CHI 2013 Mobile Accessibility Workshop collocated with CHI'13 | April 28, 2013 | | Paris (France) | 2013 | 5 | http://mobile-accessibility.di.fc.ul.pt/papers/mobacc2013_submission_29.pdf | |
| 26 | Quill: A Collaborative Design Assistant for Cross Platform Web Application User Interfaces | UCL, W3C | 22nd International World Wide Web Conference, | May 13th-17th, 2013, | ACM DL | Rio de Janeiro (Brazil) | 2013 | 4 | ISBN: 978-1-4503-2038-2 | Via ACM DL |
| 27 | A Computational Framework for Context aware Adaptation of User Interfaces | UCL | 7th International Conference on Research Challenges in Information Science | May 29th-31st, 2013 | IEEE | Paris (France) | 2013 | 11 | 10.1109/RCIS.2013.6577709 | Via IEEE |
| 28 | MMW-4S: a model-based approach for generating context-aware user interfaces for the mobile Web | CTIC | CASFE 2013 workshop collocated with EICS'13 | June 24th, 2013 | CEUR | London (U.K.) | 2013 | 4 | http://ceur-ws.org/Vol-1013/casfe2013_submission_6.pdf | Via CEUR |
| 29 | Current Practices on Model-based Context-aware Adaptation | UCL, W3C | CASFE 2013 workshop collocated with | June 24th, 2013 | CEUR | London (U.K.) | 2013 | 7 | http://ceur-ws.org/Vol-1013/casfe2013_submission_4.pdf | Via CEUR |

| | | | | | | | | | | |
|----|--|------|---|------------------|--------|---------------|------|---|---|------------|
| | | | EICS'13 | | | | | | | |
| 30 | Collaborative Working Solutions based on Web Technologies powered by the Webinos platform | TID | CASFE 2013 workshop collocated with EICS'13 | June 24th, 2013 | CEUR | London (U.K.) | 2013 | 4 | http://ceur-ws.org/Vol-1013/casfe2013_sub_mission_2.pdf | Via CEUR |
| 31 | T-OMT: A Novel Opinion Mining Tool for Improving Global Customer Relationship Management | TID | CASFE 2013 workshop collocated with EICS'13 | June 24th, 2013 | CEUR | London (U.K.) | 2013 | 4 | http://ceur-ws.org/Vol-1013/casfe2013_sub_mission_5.pdf | Via CEUR |
| 32 | Towards an Architecture Supporting Social, Adaptive and Persuasive Services for Active Elderly | ISTI | CASFE 2013 workshop collocated with EICS'13 | June 24th, 2013 | CEUR | London (U.K.) | 2013 | 6 | http://ceur-ws.org/Vol-1013/casfe2013_sub_mission_8.pdf | Via CEUR |
| 33 | An Eclipse-Based Authoring Tool For Developing Model-Based Adaptive Service Front-Ends | SAP | CASFE 2013 workshop collocated with EICS'13 | June 24th, 2013 | CEUR | London (U.K.) | 2013 | 4 | http://ceur-ws.org/Vol-1013/casfe2013_sub_mission_7.pdf | Via CEUR |
| 34 | Enhancing Collaborative Sketching with Adaptation Guidelines | UCL | CASFE 2013 workshop collocated with EICS'13 | June 24th, 2013 | CEUR | London (U.K.) | 2013 | 4 | http://ceur-ws.org/Vol-1013/casfe2013_sub_mission_1.pdf | Via CEUR |
| 35 | A Context-aware Dialog Model for Multi-device Web Apps | CTIC | Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems within EICS'13 | June 24–27, 2013 | ACM DL | London (U.K.) | 2013 | 4 | 10.1145/2480296.2480341 | Via ACM DL |
| 36 | An Environment for Designing and Sharing Adaptation Rules for Accessible Applications | ISTI | EICS 2013 | June 24–27, 2013 | ACM DL | London (UK) | 2013 | 6 | 10.1145/2480296.2480323 | Via ACM DL |

| | | | | | | | | | | |
|----|---|----------|---|---|-----------------|------------------------|------|----|---|--------------|
| 37 | Framework for the declarative implementation of native mobile applications | CTIC | IET Software | 24 July 2013 | IET | - | 2013 | - | doi: 10.1049/iet-sen.2012.0194 | No |
| 38 | Using Standards to Build the DIMAG Connected Mobile Applications Framework | CTIC | Computer Standards & Interfaces | 24 August 2013 | Elsevier B.V. | - | 2013 | - | http://dx.doi.org/10.1016/j.csi.2013.08.007 | No |
| 39 | Generation of Multi-Device Adaptive MultiModal Web Applications | ISTI | MobiWIS 2013 - International Conference on Mobile Web Information Systems | August 26-29, 2013 | LNCS - Springer | Paphos (Cyprus) | 2013 | 15 | 10.1007/978-3-642-40276-0_17 | Via Springer |
| 40 | Simplifying the Development of Cross-Platform Web User Interfaces by Collaborative Model-based Design | UCL, W3C | ACM SIGDOC'2013 | September 30 th , October 1 st , 2013 | ACM DL | Greenville - NC (USA) | 2013 | 10 | http://dx.doi.org/10.1145/2507065.2507067 | Via ACM DL |
| 41 | Enhancing Collaborative Sketching Activities with Context-aware Adaptation Guidelines | UCL | WebMedia' 2013 | November 5-8, 2013 | ACM DL | Salvador - BA (Brazil) | 2013 | 4 | http://dx.doi.org/10.1145/2526188.2526220 | Via ACM DL |

| TEMPLATE A2: LIST OF DISSEMINATION ACTIVITIES | | | | | | | | |
|---|---|-------------|---|---------------------------|-------------------------|-------------------------------|------------------|---------------------|
| NO | Type of activities ⁵ | Main leader | Title | Date | Place | Type of audience ⁶ | Size of audience | Countries addressed |
| 1 | Conference Keynote speech: Distributed User Interfaces: How to Distribute User Interface Elements across Users, Platforms, and Environments | UCL | CEDI'2010 (Interaccion'2010) | September 7-10, 2010 | Valencia (Spain) | Academic | ~300 | Europe |
| 2 | Mobile HCI 2010 | CNR-ISTI | Course: Adaptation and Continuity in Multidevice Environments | September 7-10, 2010 | Lisbon (Portugal) | R&D, Academic | ~20 | European |
| 3 | ICT Event (European Commission Information Society) 2010 | UCL | ICT 2010 | September 27th-29th, 2010 | Brussels (Belgium) | IT Industry | | Europe |
| 4 | Conference: Towards a toolkit for distributed User Interfaces | UCL | Conference at University of Elche | November 23rd, 2010 | Elche (Spain) | Academic | | Spain |
| 5 | Mobile Forum 2010 | UCL | Mobile Forum 2010 | December 8th, 2010 | Vieux Genappe (Belgium) | Industry, R&D | | Belgium |
| 6 | ICWE 2011 | UCL | Course: Multi-dimensional Context-Aware Adaptation of | June 20th, 2011 | Paphos (Cyprus) | R&D, Academic | ~20 | European |

⁵ A drop down list allows choosing the dissemination activity: publications, conferences, workshops, web, press releases, flyers, articles published in the popular press, videos, media briefings, presentations, exhibitions, thesis, interviews, films, TV clips, posters, Other.

⁶ A drop down list allows choosing the type of public: Scientific Community (higher education, Research), Industry, Civil Society, Policy makers, Medias ('multiple choices' is possible).

| | | | | | | | | |
|----|--|---------------------|--|-------------------------------|--------------------------|---|-------|----------|
| | | | Web Applications | | | | | |
| 7 | IFIP INTERACT 2011 | UCL | Course: Context-aware Adaptation of User Interfaces | September 5th, 2011 | Lisbon (Portugal) | R&D, Academic | ~20 | European |
| 8 | IFIP INTERACT 2011 | ISTI | Course: User Interface Design, Adaptation, and Migration for Multi-Device Environments | September 6th, 2011 | Lisbon (Portugal) | R&D, Academic | ~20 | European |
| 9 | SFE CWG (Service Front-End Collaboration Working Group) | TID, ISTI | Annual Meeting; "Internet of Services – Technical Collaboration Meeting for FP7 Projects"; | September 29th, 2011 | Brussels (Belgium) | R&D | | Europe |
| 10 | Standardisation CWG: "Internet of Services | W3C, TID | Technical Collaboration Meeting for FP7 Projects"; | September 29th, 2011 | Brussels (Belgium) | R&D | | Europe |
| 11 | Model Driven Day in Paris (conference, slide shows and product demos on the theme « Model Driven products») www.mdday.fr | W4 | Model Driven Day in Paris | November 24th, 2011 | Paris (France) | Industry (Architects, developers and project leads) | ~200 | France |
| 12 | Mobile World Congress 2012 | W3C | Dissemination and newsletter distribution at W3C stand | February 27th-March 1st, 2012 | Barcelona (Spain) | Industry, Research | ~1000 | Global |
| 13 | Model-based User Interfaces Working Group | W3C,UCL, ISTI, CTIC | 1st Face2Face Meeting | February 9-10, 2012 | Kaiserslautern (Germany) | Industry, Research | ~15 | Global |
| 14 | Documation (Exhibition about digitization technologies and document management systems) www.documation.fr | W4 | Posters, brochures | 20-21, March, 2012 | Paris (France) | Industry (Business users and CTOs) | ~5000 | European |
| 15 | SAP Handelssymposium 2012 | SAP | A context aware adaptive UI for warehouse picking | April 25-26, 2013 | Frankfurt (Germany) | Industry | ~200 | German |

| | | | | | | | | |
|----|---|---------------------------|---|-----------------------|--------------------|------------------------------------|--------|---------------|
| 16 | ACM-CHI Conference | ISTI | Course: User Interface Design and Adaptation for Multi-Device Environments; | May 5-10, 2012, | Austin (USA) | R&D, Academic | ~20 | Global |
| 17 | Conference Keynote speech: RCIS'2012 (Sixth International Conference on Research Challenges in Information Science) | UCL | Animated Transitions for Empowering Interactive Information Systems | May 18th, 2012 | Valencia (Spain) | Industry, Research | ~200 | European |
| 18 | Mobile Web and HTML5 Workshop | W3C, CTIC | Presentation of MyMobileWeb platform | June 6, 2012 | Barcelona (Spain) | R&D, Standardization Bodies | | European |
| 19 | Model-based User Interfaces Working Group | TID, W3C, UCL, ISTI, CTIC | 2nd Face2Face Meeting | June 14-15, 2012 | Pisa (Italy) | Industry, Research | ~15 | European |
| 20 | Solutions DEMAT' (Posters, brochures) www.salons-solutions.com | W4 | Exhibition about digitization technologies | October, 2012 | Paris (France) | Industry (Business users and CTOs) | ~20000 | European |
| 21 | II Congreso Internacional de Ingeniería de Software – | UCL | Keynote: A Review of Animation Techniques for User Interface Design | October 10th, 2012 | Quindio (Colombia) | Academic | ~300 | Latin America |
| 22 | Mobility for Business (Posters, brochures, Slide show) www.mobility-for-business.com | W4 | Exhibition about the latest technologies for mobile devices | October 9-10, 2012 | Paris (France) | Business, Integrators | ~3000 | Europe |
| 23 | ERCIM News, vol. 88 pp. 37 – 38. | ISTI | Cross modality adaptation of service front ends | 2012 | ERCIM EEIG | R&D | | |
| 24 | Model-based User Interfaces Working Group | W3C,UCL, ISTI, CTIC | 3rd Face2Face Meeting | October 29-30rd, 2012 | Lyon (France) | Industry, Research | ~20 | Global |
| 25 | Mobile World Congress 2013 | W3C, TID | Dissemination and newsletter distribution at W3C stand | 25-28 February, 2013 | Barcelona (Spain) | Industry, Research | ~1000 | Global |

| | | | | | | | | |
|----|---|----------------------|---|---------------------------------------|------------------|------------------------------------|--------|---------------|
| 26 | Keynote Speech 'II Conferencia Internacional de Ciencias Computacionales e Informáticas' http://www.informaticahabana.cu/nod/e/3456 | UCL | Informática'2013 is the 15th International Convention and Fair | March 18th to 22 nd , 2013 | Havana (Cuba) | Academia | ~250 | Latin America |
| 27 | Documation (Exhibition about digitization technologies and document management systems) www.documation.fr | W4 | Posters, brochures | 20-21, March, 2013 | Paris (France) | Industry (Business users and CTOs) | ~5000 | European |
| 28 | ACM-CHI Conference 2013 | ISTI | Course: User Interface Design and Adaptation for Multi-Device Environments; | April 29, 2013, | Paris (France) | R&D, Academic | ~20 | Global |
| 29 | FIA Dublin 2013 www.fi-dublin.eu | ISTI, TID | 2013 Future Internet Assembly event | May 8-10 th , 2013 | Dublin (Ireland) | Academia, Industry | ~300 | European |
| 30 | Model-based User Interfaces Working Group | W3C, UCL, ISTI, CTIC | 4th Face2Face Meeting | July 10th, 2013 | Munich (Germany) | Industry, Research | ~20 | Global |
| 31 | HCI International 2013 | ISTI | Course: User Interface Design and Adaptation for Multi-Device Environments; | July 23, 2013 | Las Vegas, (USA) | R&D, Academic | ~20 | Global |
| 32 | MobiWIS 2013 - International Conference on Mobile Web Information Systems | ISTI | Generation of Multi-Device Adaptive MultiModal Web Applications | August 26-29, 2013 | Paphos (Cyprus) | R&D, Academic | ~80 | Global |
| 33 | Solutions DEMAT' (Posters, brochures) www.salons-solutions.com | W4 | Exhibition about digitization technologies | October, 2013 | Paris (France) | Industry (Business users and CTOs) | ~20000 | European |