

Semantic Brokering of Multimedia Contents for Smart Delivery of Ubiquitous Services in Pervasive Environments

Alba Amato, Beniamino Di Martino, Salvatore Venticinquè
Department of Information Engineering,
Second University of Naples, Aversa, Italy

Abstract — With the proliferation of modern mobile devices having the capability to interact each other and with the environment in a transparent manner, there is an increase in the development of those applications that are specifically designed for pervasive and ubiquitous environments. Those applications are able to provide a service of interest for the user that depends on context information, such as the user's position, his preferences, the capability of the device and its available resources. Services have to respond in a rational way in many different situations choosing the actions with the best expected result by the user, so making environment not only more connected and efficient, but smarter. Here we present a semantic framework that provides the technology for the development of intelligent, context aware services and their delivery in pervasive and ubiquitous environments.

Keywords — Intelligent Agents, Context Awareness, Mobile Augmented Reality.

I. INTRODUCTION

WITH the proliferation of modern mobile devices having the capability to interact each other and with the environment in a transparent manner, there is an increase in the development of applications specifically designed for pervasive and ubiquitous environments[14]. These environments are characterized by the presence of devices such as sensors, actuators, RFID tags, etc... which make them vastly interconnected, with distributed intelligence. Thus, context-aware applications are able to provide a service of interest for the user that depends on context information, such as the user's position, his preferences, the ability of the device and its available resources (memory, cpu, peripherals). Moreover these services have to be able to respond in a rational way in many different situations choosing the actions with the best expected result, so making environment not just more connected and efficient, but smarter. This leads, as a result, the improvement of the quality of life.

Location awareness is relevant in various fields, in order to provide more efficient services to the user. Popular examples provide support to the users on a visit of a museum or of an

archaeological site, for shopping at the supermarket, on the go within a city, to a nurse for the administration of drugs, etc...

Here we present a semantic framework that provides the technology for the development of intelligent, context aware services and their delivery in pervasive and ubiquitous environments. Besides semantic-based techniques have been used for dynamically and efficiently data processing, retrieval and delivery. In order to provide a pervasive environment with advanced smart capabilities services, we integrated those technologies that enable objects of the real world to become part of the *Internet of Things*.

The paper is organized as follows: Section II introduces related work. The problem statement is described in Section III. In Section IV the framework is described and in Section V two case studies are presented. In the end conclusion and future work are discussed.

II. RELATED WORK

Many research contributions focus on the development of software/hardware architecture and frameworks for mobile context-aware applications, also based on agent technology and semantic techniques.

Context Broker Architecture (CoBrA) project from Chen et al. [1] is an agent based architecture for supporting context-aware systems in smart spaces (e.g., intelligent meeting rooms, smart homes, and smart vehicles). Central to this architecture is an intelligent agent called context broker, which maintains a shared model of context on the behalf of a community of agents, services, and devices in the space, and provides privacy protections for the users in the space, by enforcing the policy rules that they define. However their approaches did not address various characteristics of context information such as classification and dependency. The Service-Oriented Context-Aware Middleware (SOCAM) project introduces by Gu et al. [2] is an architecture for the building and rapid prototyping of context-aware services. It provides efficient support for acquiring, discovering, interpreting and accessing various contexts to build context-aware services. The main contribution of this work is the use of an ontology that allows for the description of the context in a semantic way that is independent of programming language. However the

implementation can provide and process only contextual information that is associated the meeting domain.

In particular, several frameworks for development and delivery mobile context-aware applications are used in the domain of cultural heritage and recommendation systems. CIMAD [3] (Common Infrastructure/Context Influenced Mobile Acquisition and Delivery of cultural heritage data) is a framework supporting the development of cultural heritage “services” designed and implemented within the EU Network of Excellence EPOCH (Excellence in Processing Open Cultural Heritage). CIMAD aims to address a wide range of context-aware and multi-channel services in the cultural heritage domain [4], i.e. data acquisition, content delivery, monitoring, and management. The innovative aspect of CIMAD is that it helps the development team in mastering the complexity of building, customizing and integrating site specific cultural heritage services, thus making the different implementation activities more productive and cost-effective. This framework has many similarities with our work, but it does not focus on the backend support for experts in the domain of the Cultural Heritage in order to augment the archaeological site as an editor of that site or a guide for the tourist (i.e. map editing, semantic annotation and localization of multimedia contents).

The EasiShop framework [5] assists consumers with their shopping experience. The Shopping Agent is a mobile agent that is responsible for managing the shopping list and finding the best suited products that match consumers preferences, interacting with the retailers Retailer Agents that is hosted by the wireless device in the shops. Shopping Agent is able to cross-compare the products and their prices so making informed decisions and deriving recommendation that could be passed to the consumer. In our approach semantic is used for knowledge representation and management, but the ontology is designed by experts of the application domain. It is used both as a common vocabulary to support interoperability among heterogeneous remote services, local applications and users, and for intelligent discovery of media contents. Besides we also provide tools for experts in the domain of the application in order to augment the market (i.e. map editor, semantic annotator).

III. PROBLEM MODEL

The formal objective of the framework is to define a model for delivering contents and services that provide the greatest utility for the user according to the knowledge that can be inferred using pervasive devices. The autonomic decision process is driven both by an updating process of the knowledge itself and by a set of rules that are defined by the expert of the application domain. The problem is modeled as an event driven system described by the following set of parameters:

-- A set of Events $E = \{(e_1, t_1), \dots, (e_{ne}, t_{ne})\}$ represent perceptions about the environment, detected by the device at a certain time;

-- A set of Believes $B = \{b_1, \dots, b_{nb}\}$, to be used for describing the knowledge in the application context is described by a domain ontology. The ontology has to be defined for the specific use case, by an expert of the application domain;

-- The user knowledge at the time t , about the environment and about his own preferences, is described by a set of concepts and individuals belonging to the ontology B .

For each user i we have a profile $P_i \subset B$;

-- A function $F : (E, P) \rightarrow P'$ updates the user's profile when new events have been notified to the system;

-- A set of goals: $G = \{g_1, \dots, g_{ng}\}$, whose achievement improves the user's utility;

-- A set of actions: $A = \{a_1, \dots, a_{na}\}$ can be performed by the system to achieve the goals, so increasing the user's utility;

-- A set of triggers, one for each goal:
 $\{\vec{w}_1(P), \vec{w}_2(P), \dots, \vec{w}_{nw}(P)\}, \vec{w}_i: P \rightarrow B^{na}$.

W is a $[nw, na]$ matrix of boolean values, that must be defined to start a number of actions, when some believes of the user's profile have been updated. A trigger is computed if the correspondent goal needs to be achieved. $A^T = A^T W^T W I$, where I represents a $[na, nw]$ diagonal matrix of boolean values and A^T is the resulting subset of action to be started.

For a chosen case study we have to define the set of perceptions, the ontology, the available actions, and the set of triggers. Let us define a multi-objective metric for the user utility as $U(G)$, that is the measure of the achievement of each goal. Objective of the framework is the maximization of such utility executing the minimum subset of actions that help to achieve the active goals.

The set of triggers must satisfy the following condition:
 $U(G(A)) > U(G(A \cup a_j)) \forall j: a_j \subset A - A'$.

In fact if an action cannot improve the user's utility, it can bother the user and at least it affects the performance of the system. In a general scenario the function F can be implemented by an expert system, but if $E \in B$ it simply updates the user profile by adding, removing or replacing perception in the user's profile P .

In the following we introduce a technological framework that allows for implementing the proposed model and two case studies.

IV. THE FRAMEWORK

In Fig. 1 the architectural solution of the framework is shown. The framework is composed of different tools and applications.

On the left side of Fig. 1, the user is using his device that hosts a light agent that is able to perceive information from the field by pervasive sensors. The local agent executes

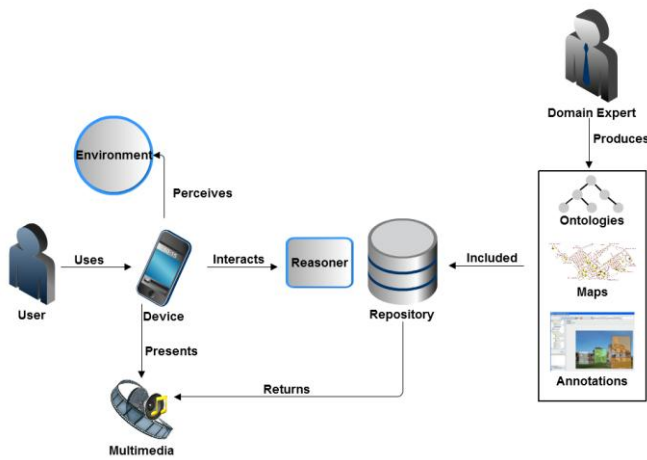


Fig. 1: Architecture and roles

autonomously and proactively in order to support the user's activity within the environment where he is moving. It discovers surrounding objects, it uses them to update the representation of the user's knowledge, reacts using the local knowledge to organize and propose the available contents and facilities by an interactive interface. If the connection works the device can access remote services, which can access a wider knowledge and have greater reasoning capabilities to look for additional contents and applications.

Remote applications implement context aware services. They use personal devices to collect perceptions and for content delivery. An ontology implements the representation of the global knowledge that is necessary to share a common dictionary and to describe the relationships among the entities/objects, which are part of the model. In our model a common ontology include all the general concepts, which are useful for a description of a pervasive environment where mobile users are moving, using their devices and interacting with available facilities and other users. The general ontology is complemented with a domain ontology that is designed by an expert of the specific application field. Experts of the application field define the domain ontology for the specific case study. They use or design a map to represent the environment. They add POIs to the map to geo-refer multimedia contents and can link them to concepts of the ontology. Furthermore they select relevant contents and annotate them using concepts and individuals of the ontology.

Concepts of the ontology are used at client side to describe a representation of the reality as it is perceived by the user. On the back-end the ontology is used to annotate digital resources like point of interests, contents, applications. It is also used to support reasoning. User's behaviors, information from pervasive devices or from other users, device properties, external events are heterogeneous data that are perceived by the device and that are used to build a dynamic changing representation of the user knowledge about the reality, within which he is moving. The applications are knowledge driven. The user's knowledge can be used by the application that is running on the device to adapt its logic locally, and it is updated remotely to improve the awareness of services at server side. Application are events based. Events can be

updates of the user's knowledge or can be explicit service requests raised by the user. At each invocation semantic queries, that depend on the user's knowledge, are built and processed to get the action to be performed and the contents to be delivered. Results of the query are individuals of the ontology that are described by semantic annotation. The user's knowledge is composed of many semantic concepts with *static*, *dynamic* or *locked* properties. Components of the knowledge are:

-- *Device technology and capability*. Among the static properties here we mean hardware resources, on board peripherals, display size, total memory and storage.

Dynamic ones can be power level, available memory and storage and bandwidth ;

-- *User's position*. It is a dynamic property that can change over the time and can be evaluated using different techniques and technologies, depending on the devices, on the available infrastructures and the kind of environment (indoor or outdoor);

-- *Pervasive objects*. They are dynamically discovered by the device. They can be sensors which provide information about the environment or can be used by the services itself by the device or eventually by thorough network if they are connected;

-- *Time information*. We intend the current time at user side and the time that the user is spending, or has spent within the environment.

-- *User's interests*. This part of the knowledge could be dynamically changed by the application according to the user's behaviors and to his feedbacks. The user could choose to start from an empty or a standard profile, to change it or to lock some properties interactively.

Semantic techniques are used for intelligent content and application discovery and delivery. Knowledge representation, ontology and annotations of digital resources are used to filter, organize and deliver contents and software to the device. Different techniques for reasoning can be experimented such as graph matching, description logics, neural networks or more simple ones like SPARQL queries. Here we integrate semantic techniques with discrete optimization methods to take into account constraints such as user's position and available time for exploitation.

V. CASE STUDIES

Here we present two different case studies, which have been chosen to test the proposed model and the developed framework.

A. Cultural Heritage

Exploitation of archaeological sites can be very difficult because of a lack of supporting infrastructures and because of the complex recognition and comprehension of the relevant ruins, artworks and artifacts[10]. The availability of personal devices can be used to plan and support the tourist by suggesting him the itineraries, the points of interest and by providing multimedia contents in the form of digital objects which can semantically augment the perceived reality. In this context relevant issues are the profiling of the user, the

discovery and the delivery of the contents that can improve the user's satisfaction, new models of interactions with reality. Three case studies have been chosen to test the approach and the framework [12]. The S. Angelo in Formis Basilica, in Campania, near S. Maria Capua Vetere, the ancient town of Norba and the amphitheater of Capua. In these sites we cannot install complex infrastructures and they are difficult to be understood without a tourist guide. For the presented case studies we need to provide a technological solution that does not need infrastructures for letting the software know the user location and his feeling about the environment. It means that Bluetooth, RFID, GPS, electronic compass, camera, network connection and others are the technologies which have to be used together or independently to get information about the user perceptions and to augment his exploitation of the archaeological site. The first thing to be considered when modeling this kind of problem is the environment. In fact we have to model the environment where the user is moving and to reconstruct the perceptions of the user himself in order to get his particular vision about what is surrounding him. A real representation of the environment is necessary to identify landmarks and possibilities of intervention using pervasive actuators and sensors whose input will be stored in the memory of the agent and will be updated as the environment changes. The environment will be modeled as a geo-referred map with itineraries, landmarks and points of interest. The user will be able to download the map of the area to be visited at home, before to leave, or on site, if the network will be available. The map will include all the points of interest that identify the relevant objects of that area and different cultural itineraries which could be exploited on site. Also contents can be discovered and downloaded in advance according to the device capabilities. On board software and remote services will assist the cultural visit by augmenting the reality by the user's personal device. A software agent executes on the user's device to support services exploitation. It perceives the surrounding environment using the on-board peripherals and executes plans which are chosen by an ad-hoc reasoning to optimize the user's satisfaction. The knowledge of the environment acquired by the agent represents part of its own beliefs. Another set of beliefs describes user's knowledge about the surrounding environment that can be acquired by using peripherals of his personal mobile device, by recording and evaluating user actions or explicitly asking for user feedbacks. Some examples are user's position, interest, nearby objects, landscape, etc. Interaction with the environment and presentation of contents can be done again using user's device and his peripherals. Of course the way of localization of users and objects depends on the device technology, the available infrastructures and the kind of environment.

Indoor or outdoor localization can be implemented using heterogeneous technologies, and often absolute localization can not be performed, but it is only possible to detect nearby landmarks or objects. Besides as the device is a PDA, it has limited battery, storage and computation capacities so also the agent running on it has limited capabilities.

In order to extend the agent's capability another remote intelligent agent interacts with the local one to achieve

individual or shared goals. This two-agent system is part of a loosely coupled network of problem-solver entities that work together to find solution to problems that are beyond the individual capabilities or knowledge of each entity [6]. In fact the agent running on the local device is able to collect perceptions, to communicate beliefs and to execute actions locally. The remote one can access a wide knowledge base and can perform a more complex reasoning. Of course different components of the user's knowledge could have different weights according to their relevance or the time of perception. We could consider the knowledge at the current time, in a time windows, or its complete history. In order to augment user's knowledge and capability to interact with the environment, services have to choose, according to their context awareness[11]; i) what content and application it has to deliver; ii) when it needs to present the content; iii) how this should be done. Finally another set of beliefs is the set of contents in the remote knowledge base which are available for delivery, to be used for augmenting the reality perceived by the user. The maximization of user satisfaction represents the goal to achieve. With this aim agents have to make a decision about the optimal set of contents and their organization according to user's profile, device and time availability.

B.Smart Market

The second proposed case study is an e-commerce service that profiles the customers and supports their purchases recommending products and shops [13]. On the other side the service is used by vendors, who upload information about their offers, shops and sales. The choice of this application field is due to the fact that the rising usage of mobiles as the primary route for consumer interaction with brands, and its importance as a catalyst for content sharing and recommendation, are still at odds with the level of focus and budget currently afforded to mobile in the market field. Additional advantages in this field brought by the technology are the traceability, which will help the consumers to verify the origins of the products. Besides, in case of a food related disease, the traceability of the purchased food will enable faster detection of the origin of the infection and thus contain its impact better and faster. A QRCode or an electronic tag can be read by the device and may carry out information related to drugs, making it easier for the customer to be acquainted with adverse effects and optimal dosage.

Besides, in case of an accident because of drug abuse or misuse it will be possible to quickly identify the taken drug by asking the smart device, which may also inform about the right antidote and dosage to enable the emergency treatment to react faster and better, and thereby saving lives.

Let us imagine users of different ages and habits, and with different purposes, in a market. Each of them has a smartphone with a smart application that perceives the environment, provides simple facilities, and accesses the e-service. All the customers continuously update, by the smart application, their profile in the system. They can insert information about themselves such as name, sex, age, home country, e-mail address, languages spoken, level of education, lists of interests, etc. They also set a few preferences about what information the device can autonomously communicate,

such as position, nearby objects, user's behaviour. Customers use a grocery list by their smartphone updating their shopping chart. Vendors add or remove their marketplaces and products, and update their description in the knowledge base, also by semantic annotations. Just to simplify our requirements, we suppose that customers cannot add, describe and recommend products on their own. Using the autonomic behavior of the service, the objective of the system is to propose the best set of recommendations, which can help the user to improve his own utility, exploiting the available information about products, about the environment within which customers are moving and about the customers themselves.

VI. CONTEXT AWARE SEMANTIC DISCOVERY

For sake of simplicity, in both the proposed case studies we suppose that the perceptions notified by the device to the remote services are the user's location, his/her available time, an concepts of the ontology. A common goal is the maximization of the user's satisfaction by proposing the best set of contents, which are relevant to the user's interest. Each time a new perception has been received, the user's profile is evaluated in order to eventually trigger the discovery service that updates the optimal set of contents to recommend.

The semantic discovery service provided by our framework returns a set of digital objects related to POIs in the pervasive environments. Each content is annotated by concepts from the ontology and can be discovered by a SPARQL query to the content repository. The result of the query is a set of N instances of digital objects whose relevance to the user context has not been considered yet. The context awareness of the services is exploited by computing the relevance of each content to the profile p , which is the semantic representation of the context, including interest, position, etc.. The annotation and the user's profile are represented using the Vector Space Model (VSM). VSM is a model for semantic representations of contents as vectors of items created by G. Salton [9]. In our case \vec{a}_j is the vector of concepts c_{ij} of the domain ontology.

$$\begin{aligned} &-- \vec{a}_j = \langle \{c_{1,j}, o_{1,j}\}, \{c_{2,j}, o_{2,j}\}, \dots, \{c_{l,j}, o_{l,j}\} \rangle \\ &\forall j = 1 \dots N \\ &-- \vec{p} = \langle c_1, c_2, \dots, c_m \rangle \end{aligned}$$

Sizes l and m are the number of different concepts that appear in the annotation and in the profile. If a term occurs in the annotation, $o_{k,j}$ is the number of occurrences of that term in the annotation.

Sizes l and m are the number of different concepts that appear in the annotation \vec{a}_j and in the profile \vec{p} . If a term occurs in the annotation, $o_{k,j}$ is the number of occurrences of that term in the annotation.

We defined a score $A(\vec{a}_j, \vec{p})$ to measure the relevance of an annotation \vec{a}_j to the profile \vec{p} by the following formula (1)

$$w_j = w(\vec{a}_j, \vec{p}) = \sum_{k=1}^l r_k \quad (1)$$

where

$$r_k = o_{k,j} * \sum_{i=1}^m \frac{1}{d_{k,i} + 1}$$

where $d_{k,i}$ is the minimum number of edges that connect the node representing the concept $c_{k,j}$ of the annotation to c_i of the profile. In (1), for each item $c_{k,j}$ of the vector \vec{a}_j the relevance to the profile \vec{p} is computed by adding the relevance of that concept to each concept of the profile, and by multiplying each contribution for the number of occurrences $o_{k,j}$. The relevance between two concepts is calculated by dividing 1 by the number of edges of the ontology that connect the node representing the concept c_k of the profile to $c_{k,j}$ plus 1. As a result we have a score for each annotated item that is associated to a POIs so that it is possible to order the items and the POIs according to user's preferences.

However a user has some additional constraints.

In the case of the archaeological guide the contents to be delivered can be limited by the device technology or the available time for the experience.

In the case of smart market, constraints can be the price, the distance he wants to go away, the purchase frequency, etc. For this reason it is necessary, to select the valid recommendations excluding, those that cannot be enjoyed according the current constraints. For example it is necessary that the budget does not exceed the amounts that is available to the user. Here we aim at recommending the best products that: i) maximize the score, ii) are compliant with the user's budget, iii) can be purchased without exceeding time and space limits.

This problem can be reduced to a *discrete optimization problem* that consists in searching the optimal value (maximum or minimum) of a function $f: \vec{x} \in Z^n \rightarrow R$, and the solution $\vec{x} = \{x_1, \dots, x_n\}$ in which the function's value is optimal. $f(\vec{x})$ is said *cost function*, and its domain is generally defined by means of a set of m constraints on the points of the definition space. Constraints are generally expressed by a set of inequalities:

$$\sum_{i=1}^n b_{i,j} x_i \leq a_j \quad \forall j \in \{1 \dots m\} \quad (2)$$

and they define the set of feasible values for the x_i variables (the *solutions space* of the problem). In our case:

$$w_i \geq 0 \quad \forall j = 1 \dots N, \quad W = \{\vec{b}_1, \dots, \vec{b}_m\}$$

where:

-- w_i represents a score and

-- B a set of constraints \vec{b}_i , each one composed of $N+1$ integer.

We have to compute:

$$\max \sum_{k=1}^N w_k x_k \quad (3)$$

so that

$$\sum_{i=1}^N b_{i,j} x_i \leq b_{j, N+1}$$

with $x_i \in \{0, 1\} \quad \forall i = 1 \dots N$. The goal is to maximize the value delivered. The vector \vec{x} represents a possible solution: its components x_i are 1 or 0 depending on whether the object is

included or not in the best set. To set the constraints we create a table with a column for each content and a row for each requirement. The rows of the matrix B are contain in each cell $b_{i,j} = 1$ if the requirement is necessary for the content in that column, $b_{i,j} = 0$ otherwise.

In Table II we specify price and shopping time for each product.

TABLE II
RAPPRESENTATION OF PRODUCT PROPERTIES AND USER'S CONSTRAINTS

	C ₁	C ₂	...	C _n	User
Price	10	20		15	35
Time	10	5		5	15

Furthermore in the last column of the table we have $b_{i,N+1} = 0$ the budget and the time availability of the user.

For example, the first row tells us that the user can spend not more than 30 , so it is necessary to exclude one content. The same consideration must be done for the second row. Finally we have to exclude that product that affects the affinity with a minimum penalty. Further details about the algorithm and its performance evaluation are out of the scope here.

In the following section we detail the technological choices which have been taken to implement each component of the framework.

VII. TECHNOLOGICAL IMPLEMENTATION

In the following sections, we present the techniques and technology choices, explaining their role in the overall picture. We refer to the introduced case studies to explain how the framework can be used in different fields.

A.Environment Map

To provide a description of the environment within which the user is moving we need a geo-referenced map that describes buildings, roads, bans, itineraries in the case of cultural heritage or departments, sectors, shelves, routes in the case of smart marketa and, in general, Points of Interest (POIs). We used the OpenStreetMap format to design open maps. In Fig. 2 the way we used to build the map of a supermarket is shown.

In the picture there are two different views of the JSOM tool, the image level and the geo-referenced map under construction. The user can choose to build the map following the edge of a reference image or can import other vectorial and geo-referenced formats like GPX. Paths, walls and POIs are outlined. Each point of interest represents departments, sectors, shelves, or any other entities of relevance and can be described using a list of keyvalue pairs. The tool allows to export the map in an open format that can be read and used by the client application.

A. Ontology

An ontology has been designed to describe the sites of interest and to annotate the related media. A general part includes the concepts which are common to all the class of

applications that can be modeled according to the proposed approach.

Among the others the *Time* class and its properties (*CurrentTime*, *ElapsedTime*), allow to organize and assist the user taking into account time information and handling time constraints. *Position* class and its properties allow to localize the user and objects around him. An application specific part of the ontology includes the concepts that belong to the domain in question.

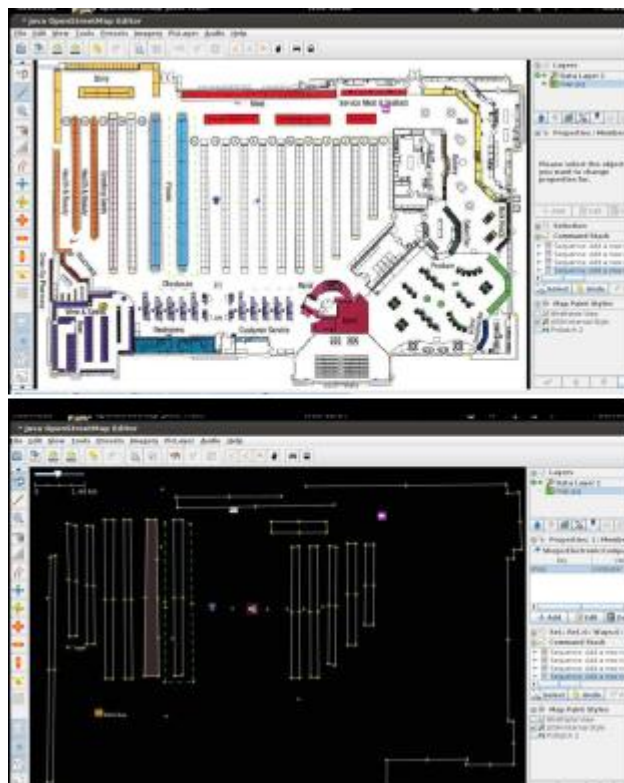


Fig. 2: Marketplace map

In the construction of ontology for the case of the archaeological guide it has been developed by experts of the cultural heritage, and we avoid here to go in deep on details which are out of our knowledge. In the case of the smart market case study, we used the following classification based on the buying habits of consumers [7]. In fact an appropriate classification allows to make better recommendations, since it is possible to identify the type of purchase made by the customer, the frequency he buys a certain product, and on the other hand it is possible to make more accurate cross-sell suggestions.

-- *Convenience products* are purchased frequently by the consumer, they do not require any major effort in the process of choice and any risk of making the wrong choice. Often, their purchase is systematic, and affordability. This also facilitates impulse buying by the individual. They are commonly illustrated by commodities, *unsought* (emergency) items, and impulse products. Examples of consumer goods that fall into the convenience category include fresh produce and grocery staples, umbrellas, gum, and batteries.

-- *Preference products* are distinguished from the previous item primarily on the basis of a perceived greater risk in their purchase. The individual, mainly because of marketing efforts, however, tends to address the problem of the choice of the brand, trying to find one that can better meet his needs. The most prominent examples of preference products are in the consumer package goods industry.

-- *Shopping products* are goods that the individual acquires after careful considerations, and after a prolonged comparison among different brands. These assets may be slightly different from the others, making the price the main driver of choice, or differentiated, making their attributes to play a key role in purchasing decisions. Examples of shopping goods are clothes, and furniture for end consumers, and equipment and components parts for industrial users.

-- *Speciality products* are goods with special features, often characterized by prestigious brands and very high prices. Examples of speciality goods include vintage imported wines, expensive sports cars, and paintings by well-known artists.

-- *Unknown products* are not known by the consumer, or the consumer would not show any interest if it were not informed about them through advertising.

In [8], three product types are identified: *convenience*, *shopping*, and *speciality* and three steps of purchase/consumption are considered: *pre*, *during*, and *post* transaction. In Table 1 each stage is further divided into areas, in which consumers may seek benefits. Information discovery is more important in the case of shopping, rather than convenience products. Each cell in the upper half of the table depicts whether that particular benefit is important in the context of the given product type.

TABLE I
PRODUCT CLASSIFICATION: THE MATCH BETWEEN PRODUCT TYPES AND MEDIA ATTRIBUTES

	Pre-Transaction		During Transaction	
	Information Search	Product Trial	Payment Options	Immediate Delivery
<i>Convenience products</i>	Not Important	Not Important	Important	Important
<i>Shopping Products</i>	Important*	Important	Important	Moderately Important
<i>Speciality Products</i>	Important**	Important	Important	Moderately Important
	Post-Transaction			
	Installation Service	Consumption	Return	After Sales Enhancement
<i>Convenience Products</i>	Not Important	Important	Not Important	Not Important
<i>Shopping Products</i>	May be Important	Important	Important	May be Important
<i>Speciality Products</i>	May be Important	Important	Important	May be Important

Since the goal here is not to design a complete profile, for the creation of the ontology we focused on general concepts that make it possible to achieve the successful personalization we aim for. We identified two main disjoint classes: *Product*

and *Customer*. The *Customer* class contains personal information and has two disjoint subclasses: *Member* and *Non member* to distinguish the users that are regular customers enrolled from the others. Personal information, inserted as datatype properties, consists of information common for all users that is useful to identify and partially segment users. Datatype properties included are *hasName* (type String), *hasAge* (type int), *hasEmail* (type String) and *hasGender* (type Man or Woman). In Fig. 3 a picture the relevant concepts of the ontology for a simple use case of Smart Market are shown.

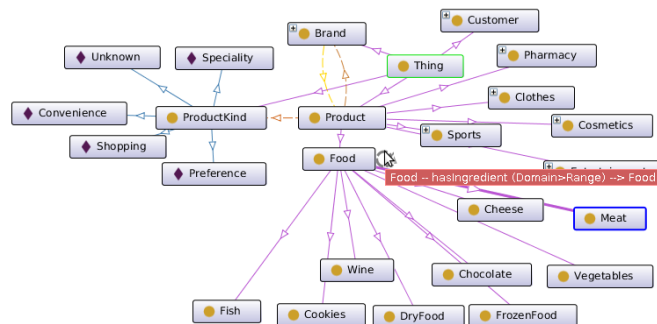


Fig. 3: Smart Market Ontology

The *Product* class has six subclasses: *Food*, *Cosmetics*, *Sports*, *Clothes*, *Entertainment*, *Pharmacy*.

The *isProductType* object property supports a classification based on the purchasing habits of consumers according to the model explained before. Besides *Food* class has the data properties: *isGlutenFree* (type boolean) indicates the presence of gluten, *hasExpirationDate* (type Date Time) registers the sell-by date expiration and *isFrom* (type string) provides information about the production chain. Another part of the ontology describes the type of purchase: *Convenience*, *Preference*, *Shopping* or *Speciality*. In particular Convenience products are characterized by the *hasPurchaseFrequency* that can be instantiated for each couple (*product*, *user profile*).

B. Annotation

The ontology is used also for annotating the multimedia contents. To annotate texts, images and any kind of contents we chose the ActiveMedia tool.

In Fig. 4 a picture of the Amphitheater of S. Maria Capua Vetere is annotated with the *Column* and the *Arc* classes which are part of this kind of building. The output produced by the annotator is an RDF file with concepts and properties of the ActiveMedia ontology and of the domain ontology. The same has been done for the smart-market, where images show those products to be recommended.

C. Digital Repository

The Fedora repository is used to store digital objects and supports their retrieval. Into the Fedora repository a digital object is composed of a set of files which are:

- *object metadata*: used by the client application to understand how to deliver the content;
- *binary streams*: which are images, video, text ... any kind of raw information to be delivered;

- *RDF annotation*: that describe the semantic of the object according to the ontology;
- *disseminations*: filters to be eventually used for adapting the object according to the target client.

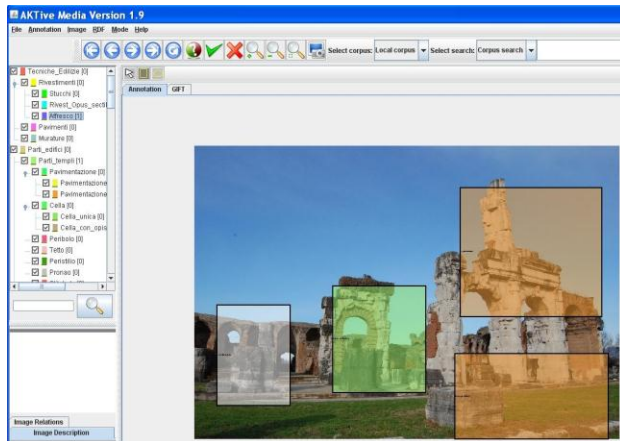


Fig. 4: The annotator

We loaded the Aktive-Media ontology and the domain ontology into the Fedora repository in order to exploit its embedded SPARQL engine that is used to select the optimal set of individuals (i.e. contents). Multimedia contents are automatically stored into the repository after the annotation phase. The RDF output is automatically processed using an XSL transformation to make it compliant with the model used by the Fedora repository.

D. Content's Delivery for the Mobile Archaeological Guide

The diversity of media content to deliver and their quality, is especially relevant in the case of the archaeological guide. In this case we have defined the following content models.

- Multiple images whose transparency can be graduated by the user to compare changes in different periods. In the same way real pictures can be compared with paintings. Old picture can be compared with what is seen by the camera.

- Parts of the image acquired by the camera are recognized and linked to related multimedia contents.

- Virtual reconstructions which are synchronized with the camera output or the detected RFIDs.

- Text, audio, video and composite media. A content descriptor is attached to every digital object. It is used by the device when the content is being delivered. The descriptor defines the right player for that media, configuration parameter and necessary input. In Fig. 5 an example of delivered content is shown. The user focuses the camera on a particular view of the S. Angelo in Formis Basilica and sees the original temple of *Diana Tifatina*. In particular the perspective viewed by the camera is the same shown by the video.

E. The Client for Mobile Archaeological Guide

At client side we extended the android version of an open source software navigator called Navit. The navigator provides basic facilities for map visualization and to guide the user along some itineraries by using the on-board GPS receiver, or

other technologies. By new extension the guide is able to sense the environment by the available peripherals, to understand the situations according to which the visit will be adapted, and to enhance user's experience.

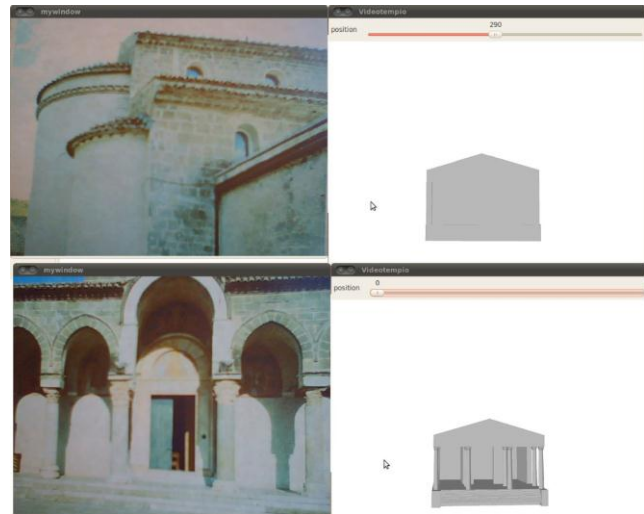


Fig.5: Synchronized video

Even if many experiences on augmented reality are proposed nowadays, usually the exploitation of vision is used only to overlap real and virtual images. Our extensions provide the following functionalities:

- access to device peripherals to sense the environment and to update the representation of the user's knowledge about the reality around him;

- access to remote services to update the user knowledge and to ask for available contents

- a local cache of objects for exploiting the visit without connection;

- a limited reasoner that is able to organize the content by itself when the connection does not work.

Perceptors are implemented by:

- GPS positioning to localize the user in open spaces and to guide him on cultural itineraries;

- RFID for positioning and detection of nearby POIs. This technology can be used to alert the user but also for his positioning in indoor environments;

- CODBAR recognition to get information about artifact, monuments, ... when a RFID reader is not available;

- image recognition by search by sample techniques which are speed-up and improved using a position based filtering;

- monitoring of device resources and configuration;

- collection of user interest by feedback and by an analysis of his behavior;

- time monitoring.

On the other hand a list of functionalities are provided to deliver contents and to guide the visit. Content management (discovery, organization and fruition) is supported both at client side and at server side. Knowledge visualization and management are provided to allow the explicit specification of user's own interest. The output of the camera is used as a component of the user's knowledge as well as a map on which

semantic additions are anchored, not simply superposed. The user will be able to ask for and exploit available multimedia contents, which are related to points of interest, or to personal interests expressed by semantic concepts. Multimedia contents will be adapted at server side according to the device/user/session profile to provide to the user the best quality of service. Fig. 6 shows the result of the content discovery service. The service invocation starts the camera that is used to take a photo of the landscape or of a subject of interest. In background, the client uploads the image and waits for the content retrieval. The discovery of relevant contents and the download of retrieved information runs in background meanwhile the user is interacting with other facilities provided by the client. At server side the picture is received together with the user's position. The received sample image is processed to extract a list of properties, which are invariant with the color, the zoom and little rotations by the OpenCV library. The server owns a list of images of landscapes and relevant points of interest the user can shoot. These images have already been processed by the same OpenCV library. The user's position is used to perform a filtering of the points of



Fig. 6: The Client

interest close to him, in order to speed-up the comparison of sample features with the ones of candidate images. The best matching images are returned. If they have been annotated, annotations are used to suggest other contents that are semantically related. Furthermore frames of the images, which have been semantically annotated, can be discovered and pointed in the picture that has been just taken. Other events, which open dialog for suggesting actions, itineraries, POIs or media can be related to new perceptions.

F. The Client for Smart Market

To access the smart market the user will exploit, by his smartphone, an Android application using an interface shown in Fig. 7 that is able to get and communicate his position and to localize POIs or entities close to him. Furthermore the user is able, by the application, to search for products in a grocery list and to add the products to be bought in his shopping cart. Let us suppose that the user wants to buy cookies. Its device updates periodically a remote agent about the user's position. Each time this component of the user's profile changes, the

agent checks for a POIs that is linked to a product semantically related to the cookies concept.

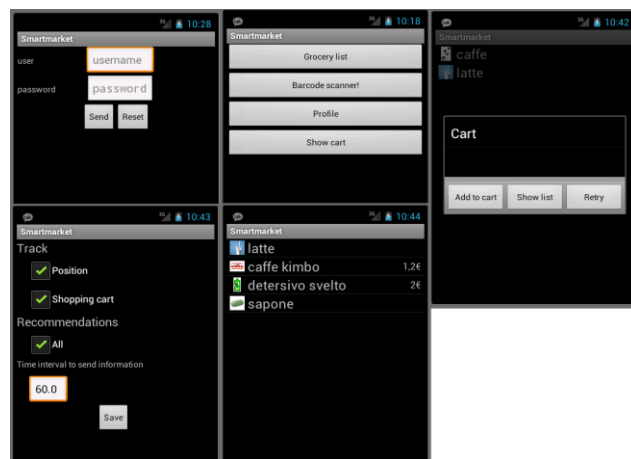


Fig. 7: The Android Client

If the POIs is found the user is notified. Starting from the POI the user can browse the list of products, which are sold at that position. In the same scenario we can imagine that the user has bought the product. Another client application of our smart service is able to recognize the user, by a fidelity card, or by any kind of personal identification mechanism, without getting any additional private information. Scanning the barcode of the purchased product the smart market service can be informed about new convenience goods in the user's habit. A new timer will be created to alert the user at the sellby expiration date of the cookies. The purchase frequency is not known and can be put equals to the expiration date. When the same, or an equivalent product, will be bought, the frequency timer will be updated and set again. Finally a list of concepts have been added to the user's profile for personalizing the discovery of similar products to recommend. In the provided example we will have *glutenFree*, *chocolate*, the *cookies brand*. At the expiration of the timer the user will be alerted about the eventual opportunity to buy the same cookies, but alternative products will be recommended according to the context aware semantic discovery strategy described in the following section. In particular among the available products the system recommends the one annotated with the *cookies* concept, the *hasBrand* property, the *glutenFree* property, and a *hasIngredient* property. Annotated texts and images are currently presented by the web interface shown in Fig. 8. In particular you have on the left the list of relevant concepts, and the picture of the product on the right.

At this point the user can move the mouse over the concepts and see the annotation on the picture. He can also move the mouse over the picture and see a tool-tip that explains the annotation.

Concepts from the list, or from the tool-tip, can be easily added to the user profile in order to let the system learning about the user's preferences for a personalized recommendation. Using the mouse button or the form is also possible to search for other products, which are semantically related to the selected concepts, in order to choose an alternative brand or similar kinds of cookies.

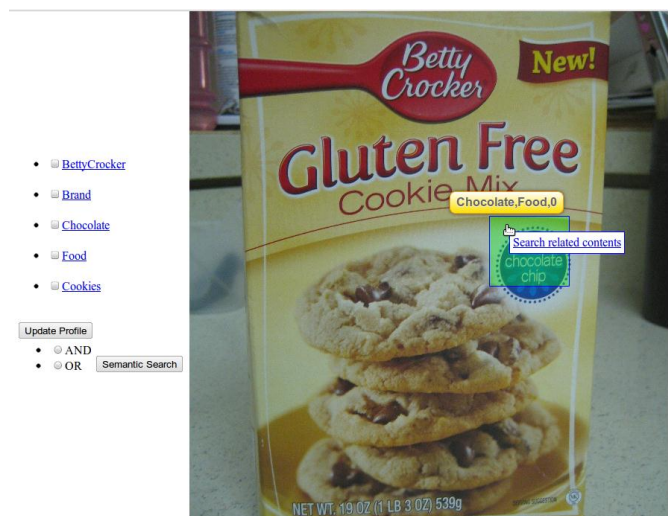


Fig. 8: Semantic browsing of a recommended product

VIII. CONCLUSION

In this paper, we proposed a proactive semantic framework for the intelligent delivery of services in pervasive environments. In such environments, users need to have access to the most relevant and interested services within a rich and dynamic context. The main goal is the exploitation of pervasive devices to maximize the expected satisfaction of the user by adapting applications and contents to be delivered and by augmenting and improving his utility in several contexts. Personal devices are used to interact with the environment, to run interactive applications and to present contents to the user. A smart application runs on the user device, uses the peripherals to perceive the environment, provides context awareness to remote discovery services. We described a framework that implements the proposed approach and described its application for cultural heritage and shopping recommendation. Future works will address quantitative analysis of the approach and the support of collaboration among users by personal semantic based recommendation.

ACKNOWLEDGMENT

Author thank Massimiliano Diodati and Vincenzo Reccia for their technical support.

REFERENCES

- [1] H. Chen, "An Intelligent Broker Architecture for Pervasive Context-Aware Systems". *PhD thesis, University of Maryland*, Baltimore County, 2004.
- [2] T. Gu, H. K. Pung, D. Q. Zhang, "A service-oriented middleware for building context-aware services", *Journal of Network and Computer Applications*, Vol. 28, pp. 1-18, Jan. 2005.
- [3] Manzaroli D., "Architectures for Context Aware Services in Smart Environments". *PhD thesis*, May 2010
- [4] G. Raffa, P. H. Mohr, N. Ryan, D. Manzaroli, M. Pettinari, L. Roffia, S. Bartolini, L. Sklenar, F. Garzotto, P. Paolini, T. Salmon Cinotti, "CIMAD - A Framework for the Development of Context-aware and Multi-channel Cultural Heritage Services", in *ICHIM07* September 30,

- 2007 Toronto
- [5] S. Kurkovsky and K. Harihar, "Using ubiquitous computing in interactive mobile marketing," *Personal Ubiquitous Computing*, vol. 10, no. 4, pp. 227-240, Mar. 2006.
- [6] V.R., Durfee, E.H., Lesser and Corkill, D.D., *Trends in Cooperative Distributed Problem Solving*, IEEE Transactions on Knowledge and Data Engineering, Vol. March 1989, KDE-1(1), pages 63-83.
- [7] E. B. Murphy, P., "Classifying products strategically," *Journal of Marketing*, vol. 50, no. 3, pp. 24-42.
- [8] B. Sindhav and A. L. Balazs, "A model of factors affecting the growth of retailing on the internet," *Journal of Market-Focused Management*, vol. 4, pp. 319-339, 1999.
- [9] G. Salton and M. E. Lesk, "Computer evaluation of indexing and text processing," *J. ACM*, vol. 15, no. 1, pp. 8-36, Jan.1968.
- [10] A. Amato, B. Di Martino, S. Venticinque "BDI Intelligent Agents for Augmented Exploitation of Pervasive Environments". In: *Proceedings of the 12th Workshop on Objects and Agents*. Università della Calabria, 2011, vol. 741, p. 81-88, ISBN/ISSN: 1613-0073
- [11] S. Venticinque, A. Amato, B. Di Martino *Ontology driven Augmented Exploitation of Pervasive Environments*. In: *AFIN2011: The Third International Conference on Advances in Future Internet*. Nice/Saint Laurent du Var, France, August 21-27/IARIA, p. 41-48, ISBN/ISSN: ISBN: 978-1-61208-148-9
- [12] A. Amato, B. Di Martino, S. Venticinque "Semantically Augmented Exploitation of Pervasive Environments by Intelligent Agents". In: *Mue 2012: The Sixth International Conference on Multimedia and Ubiquitous Engineering*. Madrid, Spain, July 10 - 13, ISBN/ISSN: ISBN: 978-0-7695-4701-5/12
- [13] A. Amato, B. Di Martino, S. Venticinque (2012) *A Semantic Framework for Delivery of Context-Aware Ubiquitous Services in Pervasive Environments*. In *INCoS 2012: 4-th International Conference on Intelligent Networking and Collaborative Systems*. Bucharest, Romania, September 19 - 21, 2012, ISBN/ISSN: ISBN: 978-0-7695-4808-1/12
- [14] Aversa, R., Di Martino, B., Venticinque, S. *Distributed agents network for ubiquitous monitoring and services exploitation (2009) Proceedings - 12th IEEE International Conference on Computational Science and Engineering, CSE 2009*, 2, art. no. 5284221, pp. 197-204.



Alba Amato is a PhD student in Computer Engineering at the Information Engineering Department of the Second University of Naples, Italy. She graduated with a Masters degree in Computer Science from the University of Napoli Federico II. Her research interests focus on intelligent systems, agent architectures, multi-agent systems and their applications in cloud computing.



Salvatore Venticinque is an Assistant Professor at Department of Information Engineering of the Second University of Naples. He received his PhD in Electronic Engineering in 2003. He has been an Adjunct Professor in Computer Programming and Computer Architecture in regular courses. He is involved in research activities dealing with parallel and grid computing and mobile agents programming for distributed systems.



Beniamino Di Martino is Full Professor of Information Systems at the Second University of Naples (Italy) since 2005. He participated to various research projects supported by national and international organizations. He is vice Chair of the Executive Board of the IEEE CS Technical Committee on Scalable Computing. His research interests include: Knowledge Discovery and Management, Semantic Web and Semantic Web Services, Semantic based Information Retrieval, Cloud Computing, High Performance Computing and Architectures, Mobile and Intelligent Agents and Mobile Computing, Reverse Engineering, Automated Program Analysis and Transformation, Algorithmic Patterns Recognition and Program Comprehension.