POLITECNICO DI TORINO

SCUOLA DI DOTTORATO

PhD Course in Computer And Control Engineering – XXVI cycle

PhD Dissertation

Context-Aware Service Creation On The Semantic Web

Oscar Rodríguez Rocha
178691

Tutor
prof. Maurizio Morisio

March 2014

Summary

With the increase of the computational power of mobile devices, their new capabilities and the addition of new context sensors, it is possible to obtain more information from mobile users and to offer new ways and tools to facilitate the content creation process. All this information can be exploited by the service creators to provide mobile services with higher degree of personalization that translate into better experiences.

Currently on the web, many data sources containing UGC provide access to them through classical web mechanisms (built on a small set of standards), that is, custom web APIs that promote the fragmentation of the Web. To address this issue, Tim Berners-Lee proposed the *Linked Data principles* to provide guidelines for the use of standard web technologies, thus allowing the publication of structured on the Web that can be accessed using standard database mechanisms.

The increase of Linked Data published on the web, increases opportunities for mobile services take advantage of it as a huge source of data, information and knowledge, either user-generated or not.

This dissertation proposes a framework for creating mobile services that exploit the context information, generated content of its users and the data, information and knowledge present on the Web of Data.

In addition we present, the cases of different mobile services created to take advantage of these elements and in which the proposed framework have been implemented (at least partially). Each of these services belong to different domains and each of them highlight the advantages provided to their end users.

Acknowledgements

First of all, I would like to thank my doctoral advisor Prof. Maurizio Morisio (with whom I've had the pleasure of working together for several years before starting the doctoral program), his valuable advice, support and teachings have allowed me to achieve my objectives in this program. Likewise, I thank my colleagues of the SoftEng group: great human beings with whom I initiated a lifetime experience three years ago, in which we have learned to collaborate and help each other.

I am grateful to all the people of the Context-Awareness research group of Telecom Italia. The research collaboration we have had, has given me the opportunity to grow as a person and as a professional. I can't thank enough Carlo Licciardi, Laurent-Water Goix and Boris Moltchanov.

Thanks also to INRIA's Wimmics group in France (especially Luca Costabello, Serena Villata and Fabien Gandon), thanks to them I had a wonderful experience in France in which I learned many essential things to achieve the objectives of my PhD.

Last but not least, thanks to my family and friends who have been supporting me at any moment and in many cases despite the distance. I'm sure that my grandparents and my aunt Hilda are proud of this achievement in heaven. I could not miss a "thank you mama", I am who I am because of you!

Contents

List of Figures

Part I

Prelude

Chapter 1

Introduction

Contemporary Web has involved end-users in such a way that they have become not only content consumers but also content producers. A huge amount of user-generated content is produced daily and it keeps continuously increasing.

With the increase of the computational power of mobile devices, their new capabilities and the addition of new context sensors, it is possible to obtain more information from mobile users and to offer new ways and tools to facilitate the content creation process. Unlike common content present over the web, content generated by users in mobility situations may include information of the context in which it was created. All this information can be exploited by the service creators to provide mobile services with higher degree of personalization that translate into better experiences.

Currently on the web, many data sources containing UGC provide access to them through classical web mechanisms (built on a small set of standards), that is, custom web APIs that promote the fragmentation of the Web. To address this issue, Tim Berners-Lee proposed the *Linked Data principles* to provide guidelines for the use of standard web technologies [25], thus allowing the publication of structured on the Web that can be accessed using standard database mechanisms.

The increase of structured data published on the web [48], increases opportunities for mobile services take advantage of it as a huge source of data, information and knowledge, either user-generated or not.

1.1 Research Context

1.1.1 The explosion of mobile User-generated Content

In the last decade Internet has grown tremendously. Current Web has involved end-users in such a way that they have become not only content consumers but also content producers. A huge amount of user-generated content is produced daily and keeps continuously increasing.

This kind of content is commonly known as User-generated content (UCG). UCG represents media content generally produced by users through diverse technologies, for example photos, digital videos, blogs, podcasts, forums, reviews of sites, and wikis. According to [83], UGC is defined as:

1. content made publicly available over the Internet
2. which reflects a certain amount of creative effort, and
3. which is created outside of professional routines and practices

Even when most users produce UGC without expecting remuneration, they are moved by motivating factors such as connecting with peers, prestige, and self-expression.

Normally, users who produce UGC are moved by motivating factors such as connecting with peers, prestige, and self-expression without expecting remuneration for that. However, UGC can be considered as a market driver for IT players by combining user-generated content with users' context data in order to create new services [67].

Nowadays, there is a huge interest in creating applications around UCG, for example complementing large-scale geospatial modeling by letting users add details and refine a 3D city model in real time using their mobile cameras [21]; in an e-tourism scenario mobile users can upload photos to an e-tourism application with their camera phones in order to send updates to a context-aware platform which automatically adds context tags to the content used to help users to filter and browse pictures and videos according different context parameters.

According to [83], the UGC can be classified in the next groups:

- Text, novel and poetry
- photo and images
- music and audio
- video and film
- citizen journalism
- educational content
- mobile content
- virtual content

With the increase of the computational power of mobile devices, their new hardware capabilities and the addition of new context sensors, the generation and consumption of UGC is moving more and more to the mobile scenario. In order to have an idea of the current dimension of the mobile User-generated Content present on the Web, we can consider the case of the mobile-based photo sharing system Instagram[1]. It was launched on October 6, 2010 and at the moment of writing it contains nearly 16 Billion of photos shared.

Another advantage of the evolution of mobile devices is that they make possible to obtain more context information from the mobile users, specifically in the situations in which they produce content. All this information can be exploited and associated to the content in such a way that it can be further used to retrieve it either by the system or by an end-user (content consumer). Unlike common content present over the web, content generated by the user in a mobility situation includes information of the context in which it was created.

1.1.2 Context-Awareness Opportunity

In Computing Science, context is defined as the set of environmental states that can determine the behavior of an application or in which an application event occurs and is interesting to the user. [75] defines context as "information that can be used to characterize the situation of an entity, where an entity can be a person, place, or physical or computational object", and context-aware as "applications that use context to provide task-relevant information and services to a user".

Context includes valuable information about the user's environment that can be represented by different variables as for example: geographical location, coordinates, surrounding points of interest (POI), nearby buddies, surrounding traffic situation, current place temperature, network connectivity, nearby resources, time of the day, weather conditions, etc.

Context-aware computing looks forward to anticipate the needs of users in order to act in advance by taking into account their context. In this way a computer accessing the context of the user can increase the richness of communication in human-computer interaction and make it possible to produce useful and innovative services [17]. By extension, context-aware applications can discover contextual information even in continuously changing environments by adapting their behaviors accordingly [84].

Moving to the field of mobile computing, [47] states that the main goal of the context is to "enable the device to better serve for people, either human computer interaction or context-aware mobile application/service".

[1] http://instagram.com

1.1.3 The Growing Web of Linked Data

The idea of extending the capabilities of the Web in order to publish structured data on it, exists from its own creation. It's in 1994 when Tim Berners-Lee[2] highlights the need of introducing Semantics in the Web to achieve this idea. Later, this idea became known as "Semantic Web".

While the Semantic Web is the goal, Linked Data provides the means to make it reality [26], it refers to a set of best practices for publishing and connecting structured data on the Web in order to increase the number of data providers and consequently accomplish the goals of the Web of Data. In this way Linked Data makes it possible to semantically interlink and connect different resources at data level regardless the structure, authoring, location etc.

The set of principles are:

1. Use URIs as names for things.
2. Use HTTP URIs, so that people can look up those names.
3. When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL).
4. Include links to other URIs, so that they can discover more things.

Data published on the Web using Linked Data has resulted in a global data space called the Web of Data. Mainly through the efforts of the scientific community and the W3C LOD project[3], more and more data have been published on the Web of Data, helping their growth and evolution.

To demonstrate this growth (since the introduction of the LOD project), Fig. 1.1 shows the data sets published on the Web as Linked Data by 2007 while figures 1.2 and 1.3 by 2009 and 2011 (most recent study at the time of writing) respectively[4].

Despite the continuous and steady growth of the Web of Data, a study conducted in 2011[5] states that it is composed of data belonging to different topical domains including User-generated Content, for which we have:

- 20 datasets out of 295
- 134,127,413 triples out of 31,634,213,770. It means only the 0.42%

[2] http://www.w3.org/Talks/WWW94Tim

[3] http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData

[4] http://lod-cloud.net

[5] http://lod-cloud.net/state/#domains

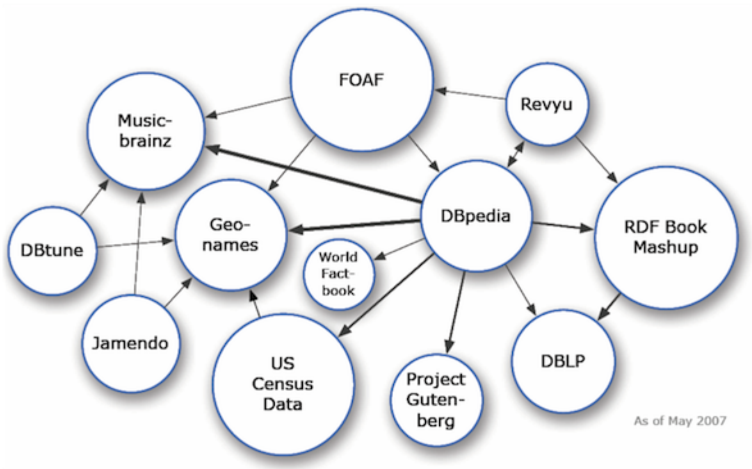

Figure 1.1. Linking Open Data cloud as of May 2007

- 3,449,143 RDF links out of 503,998,829. Thus the 0.68% of all the links

A graphical overview of the amount of triples as well as the amount of RDF links per domain is presented in Fig. 1.4 and Fig. 1.5, respectively.

While on one hand the amount of User-generated Content present in the common Web continues to grow, its presence in the Web of Data is still minor compared to the other topical domains. This situation opens up new possibilities for the creation of services and tools that enable end users to create and consume content directly from the Web of Data, as well as to extract knowledge from data that is currently present on it, as in the end the Web of Data can be viewed as a comprehensive database that can be queried at any time.

1.2 Scope of the Research

The main objective of this dissertation is to study the current case of mobile services and how they can be improved to provide more benefits to its existing and potential end users, through the inclusion of technologies that exists around them and have evolved rapidly. Some of these technologies have not been already integrated into them, opening new opportunities for creators of mobile services.

In particular, this work focuses on studying how to take advantage of the large amount of user-generated content present in the web, due to the increasing trend of users to be not only creators but also consumers of content. Likewise, how to benefit from the possibilities given by the dynamic adaptation of the mobile

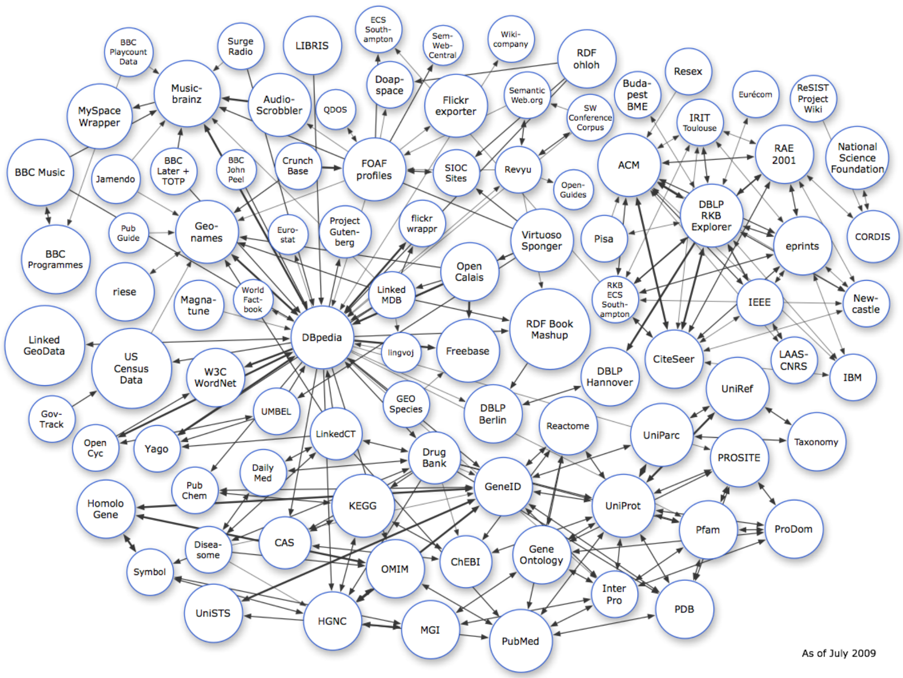

Figure 1.2. Linking Open Data cloud as of July 2009

services functionalities based on the context information of its users. Similarly, how to allow mobile end-users exploit the potential fo the Web of Data, which offers big possibilities for retrieving useful content in mobile situations.

Therefore this research work is divided into two main areas:

- The creation of a modular framework that allows mobile service creators to reuse generic modules that provide basic functionalities to create innovative mobile services, avoiding the effort of recreating existing components and rather focusing on their functionality.
- Demonstrate how to create or improve mobile in different application domains, through the complete use of the framework or the reuse of some of its components. Similarly, demonstrate the benefits obtained by the end-users through their use.

It must be stressed that the development of this work not only considers the technological knowledge and application of the various elements mentioned above, but principally focuses on the scientific research behind the problems inherent to

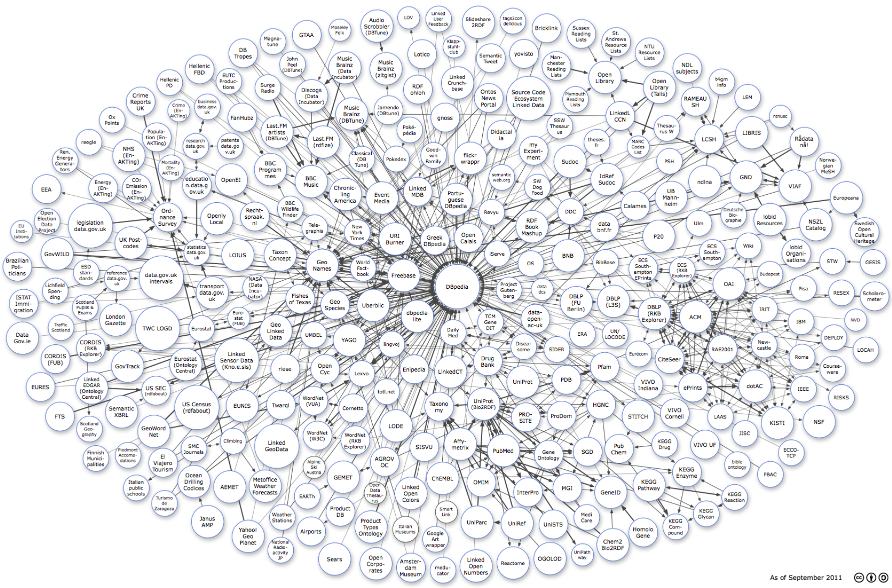

Figure 1.3. Linking Open Data cloud as of September 2011

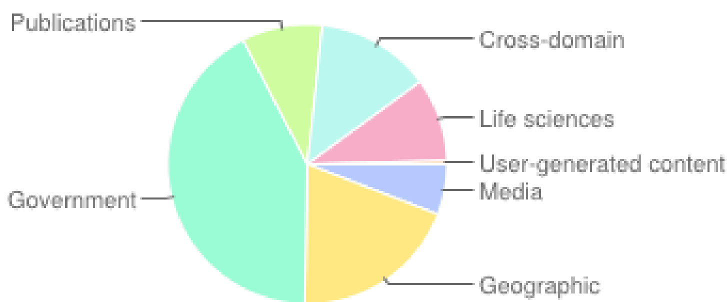

Figure 1.4. Linking Open Data distribution of triples by domain as of 2011

them. Therefore, this dissertation is presenting both aspects, thus combining the technological and scientific parts to produce a final benefit.

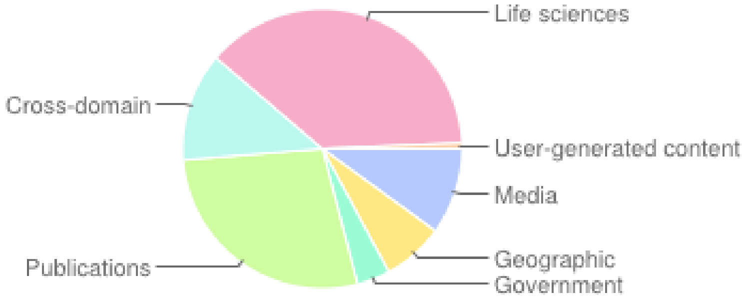

Figure 1.5. Linking Open Data distribution of links by domain as of September 2011

1.3 Outline

The present work, has been structured into three main parts: First, on *Mobile Service Creation Framework*, the research work carried out in order to build a modular framework that simplifies the creation of mobile services that take advantage of their mobile users' generated content, context information and the Web of Data is presented.

Then, part III describes the use cases of mobile services (created by using the proposed framework as a base), aimed at different applications domains, each one includes at least one of the modules listed above and shows how their inclusion has added some value to the end-users:

Chapter 3 describes a mobile service designed to provide users a way to share their personal content on the go. This service has included the modules of Semantic Annotation, ContextML-based Context Management and Content Management to provide more benefits to its users during the content generation and consumption phases. This kind of service is ideal for applications like eTourism, in which tourists can visualize information and content related to their current context.

Two mobile services that exploit the Semantic Annotation, ContextML-based Context Management and Content Management modules are described on chapter 4. One of them aims at improving the experiences of mobile users when reading eBooks through a set of social, sharing and content generation features (automatic semantically enriched notes and comments that can be further retrieved easily). The other one provides an approach to Augmented Reality based on Context-Awareness and the real-time augmented visualization of User-generated content.

Chapter 5 describes an evolution of the previous described mobile service, in this case, the ContextML-based Context Management is used in form of a Publish / Subscribe Generic Enabler (GE) and deployed into the FI-WARE EU project testbed.

Finally, chapter 6 shows a platform that aims to support emergency situations through the use of the context information of the mobile user that is in critical situation. The service incorporates the ContextML-based Context Management module.

1.4 Contribution

The main contribution of this work is the design and implementation of a framework that allows the development of innovative mobile services on top of it, which combine User-generated Content, the contextual information of their users and takes advantage of the Web of Data and the enormous knowledge present on it. Similarly, the creation of practical use-cases (within in different domains) in which such framework is used and its benefits are demonstrated. In addition to these major contributions, other more specific are listed below:

- The creation of an automatic Semantic Annotator intended for annotating the related information of User-generated Content. It is described in 2.2
- a module for handling contextual information based on semantics and offers possibilities for further control access to resources based on contextual rules in 2.1.2
- a contribution to the European project FI-WARE, by creating two Generic Enablers (GE), (one for management of contextual information an the other for the semantic annotation of information) deployed in the cloud and that can be used by different platforms in 5

Part II

Mobile Service Creation Framework

Chapter 2

A Module-based Framework

The framework presented in this dissertation, attempts to provide the creators of services with a modular system that provides the basic elements of a mobile service that integrates features like context management, user-generated content management (from the semantic annotation to the storage) and finally content retrieval that allows further use in many application domains. We are aware that due to the complexity and heterogeneity of mobile services, it would be almost impossible to cover all the necessary components to create any mobile service, that's why this work aims only to provide a set of modules that can be used either together or individually. This is possible thanks to the modular design with which it was conceived.

Figure 2.1 shows the architecture of the framework. Each of its modules will be described on detail on the next sections. As mentioned above, chapter III presents the use cases of mobile services in which the framework is implemented.

2.1 Context Management[1]

Mobile services can exploit context information obtained from its users in order to offer better personalization and experiences. This module aims to provide a solution to meet the standard needs of handling contextual information of the mobile services built on top this framework.

For the design and development of this module, it was necessary to analyze the already existing context-aware systems as well as the common principles of to

[1] **This section is partially based on [32]**

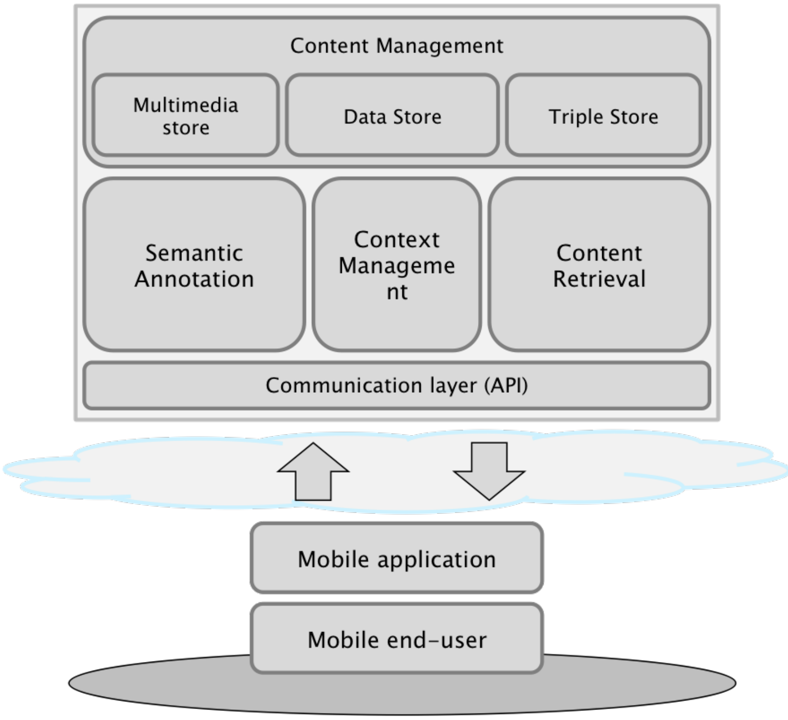

Figure 2.1. Framework Architecture

design them. In [22], the comparison of 8 different architecture approaches as well as the classification of six different context models is presented.

This module has been developed to provide two different modalities for handling context information: ContextML-based and Semantic-based.

- The first mode, offers the advantage of simplicity and lightness in the representation of contextual information, using XML[2] as a representation language and REST [40] as a means of communication. This method relies on the integration with the Telecom Italia's Context-Awareness platform to provide a set of features that will be described below.

- On the other hand, the second mode features the use of semantic technologies (such as the prissma vocabulary [10]), to represent mobile context concepts.

[2]http://www.w3.org/XML/

Its main advantage, is the possibility of controlling the access to the resources present in the platform by defining a set of rules that operate based on the context of the users. This mode is based on the HTTP-based implementation of the Shi3ld platform [33].

2.1.1 ContextML-based Context Management

Context-aware integrated platform

The main objective of this mode is to provide the mobile services using the framework, the possibility to obtain a set of ready-to-use functions for managing context information which have been previously tested. To make this possible, the Context-Awareness platform [39] of the principal Mobile Network Operator (MNO) in Italy has been integrated as a Server Enabler. The access to this service enabler is performed through a RESTlike interface.

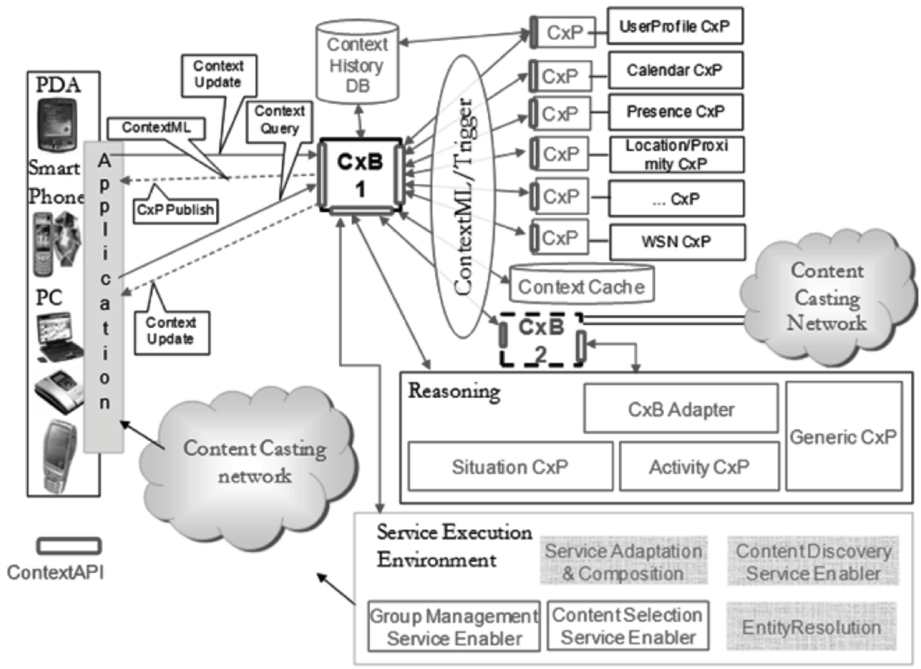

Figure 2.2. Telecom Italia's Context Management Framework

Figure 2.2 shows the architecture of the platform. It bases its events notifications, in stateless REST invocations. Its main component is the Context Broker that receives data in ContextML format. Context providers (CxP) can be plugged according to the needs of the applications, for example, calendar providers to retrieve user's events, location providers to infer more information about a specific location. Below is a more detailed description of the main components:

- **Context Consumer (CxC)** Any component that queries for and uses context data by sending a subscription request to the Context Broker.
- **Context Provider (CxP)** A component whose task is to provide context information of a certain type, e.g. weather, location, activity, etc.
- **Context Broker (CxB)** Enables the discovery of Context Providers, manages the exchange of context information between providers, sources and consumers, caches context data by implementing mechanisms of storing, retrieving, querying, deleting, and matching context elements.
- **Context cache and Context History** The context information received by the Context Broker is stored in a Context Cache while it is not expired. After its expiration it is preserved in a database managed by the Context History module.

An overview of ContextML

In this mode, the means for representing context information, context subscription/notification and some control messages is ContextML [54]. ContextML is mostly used for light weight representation of context information, ensuring fast processing also on resource constraint mobile devices. It is composed of two main elements:

- Entities: can be defined as the subject of interest which context data refers to, and it is composed of two parts: a type and an identifier
- Scopes: a set of closely related context parameters

An example of context information representation is shown in figure 2.3. In the example, some information of the user with the *username* "oscar" is represented as well as the date when it was created and period of validity *expires*. The location of the user is represented though geographical coordinates (latitude, longitude an accuracy). This location information can be further processed by the platform and by using the available Context Providers (CxP) it can be enriched with inferred additional information such as civil address, country, current type of activity being performed on that moment, etc.

```
<contextML>
<timeRef>2011-01-04T17:51:44+02:00</timeRef><ctxEls><ctxEl>
         <contextProvider id="pico-client"/>
         <entity id="oscar" type="username"/>
         <scope>position</scope>
         <timestamp>2011-04-29T14:15:39+02:00</timestamp>
         <expires>2011-04-29T14:30:39+02:00</expires>
         <dataPart>
            <par n="latitude">45.062814</par>
            <par n="longitude">7.662732</par>
            <par n="accuracy">50</par>
         </dataPart>
      </ctxEl></ctxEls>
</contextML>
```


Figure 2.3. A ContextML example

2.1.2 Semantic-based Context Management

An overview of Shi3ld

The current module features a Semantic-based Context information management, providing also the capabilities of access control to the resources. It is based on the HTTP implementation of the Shi3ld authorization model for SPARQL endpoints. Shi3ld [33] presents the following key features:

- **Attribute-based Paradigm** Shi3ld is an attribute-based authorization framework, i.e., authorization check is performed against a set of attributes sent by the client with the query that targets the resource. Relying on attributes provides broad access policy expressiveness, beyond access control lists. That means, among all, creating location-based and temporal-based access policies.
- **Semantic Web Languages Only** Shi3ld uses access policies defined with Semantic Web languages only, and no additional policy language needs to be defined. In particular, the access conditions specified in the policies are SPARQL ASK queries.
- **CRUD Permission Model** Access policies are associated to specific permissions over the protected resource. It is therefore possible to specify rules satisfied only when the access is in create, read, update and delete mode.
- **Granularity** The proposed degree of granularity is represented by named graphs, allowing protection from triples up to whole dataset.

The HTTP-based interaction with Linked Data requires some major modifications to the above features. Although we keep the attribute-based paradigm and the CRUD permission model, the new versions of Shi3ld satisfy also the following requirements:

- **Protection of HTTP Access to Resources** Protected resources are retrieved and modified by clients using HTTP methods only, without SPARQL querying.
- **RDF-only Policies** In the SPARQL-less scenario, access conditions are RDF triples with no embedded SPARQL.
- **Granularity** The atomic element protected by Shi3ld is a resource.

We rely on the definition of resource provided by the W3C Linked Data Platform Working Group: LDP resources are HTTP resources queried, created, modified and deleted via HTTP requests processed by LDP servers. Linked Data server administrators adopting Shi3ld must define a number of access policies and associate them to protected resources. Access policies and their components are formally defined as follows:

- **Definition 1.** (Access Policy) An Access Policy (P) is a tuple of the form P $= \{ACS, AP, R\}$ where (i) ACS is a set of Access Conditions to satisfy, (ii) AP is an Access Privilege, and (iii) R is the resource protected by P.
- **Definition 2.** (Access Condition) An Access Condition (AC) is a set of attributes that need to be satisfied to interact with a resource.
- **Definition 3.** (Access Privilege) An Access Privilege (AP) is the set of allowed operations on the protected resource, AP $= \{Create, Read, Update, Delete\}$.

The lightweight vocabularies used by Shi3ld are s4ac[3] for defining the policy structure, and prissma[4] for the client attributes[5]. Client attributes include user profile information, device features, environment data, or any given combination of these dimensions, in compliance with the widely-accepted definition by Dey [34] and the work by Fonseca et al.[6]. We delegate refinements and extensions to domain specialists, in the light of the Web of Data philosophy. The main classes

[3]http://ns.inria.fr/s4ac

[4]http://ns.inria.fr/prissma

[5]http://www.w3.org/2005/Incubator/model-based-ui/XGR-mbui/

[6]http://www.w3.org/2005/Incubator/model-based-ui/XGR-mbui/

and properties of these vocabularies are visualized in Figure 2.4. Shi3ld offers a double notation for defining access conditions: with embedded SPARQL (Figure 2.5a) for SPARQL-equipped scenarios, and in full RDF (Figure 2.5b), adopted in SPARQL-less environments.

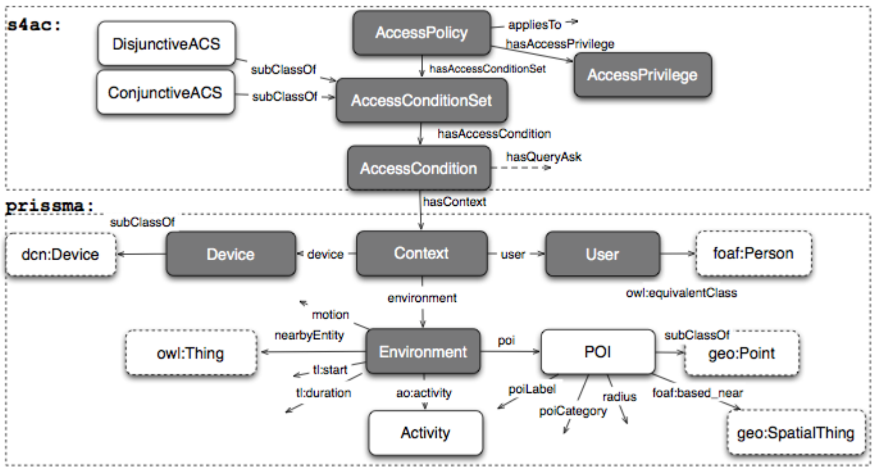

Figure 2.4. Interplay of s4ac and prissma vocabularies for Shi3ld access policies

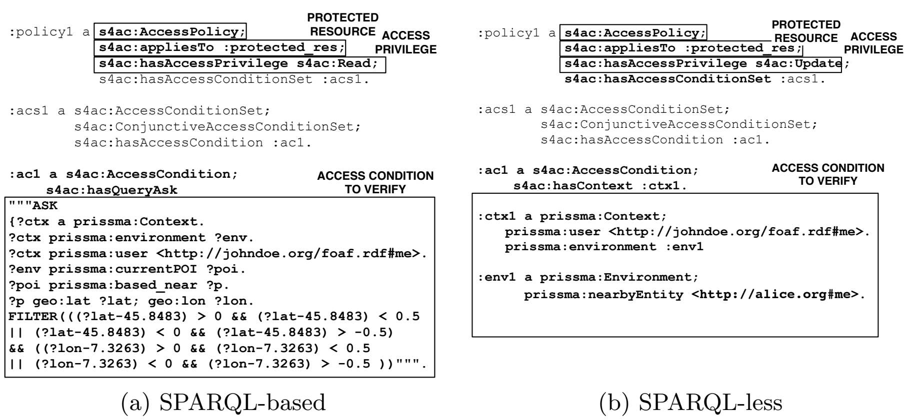

Figure 2.5. Shi3ld access policies, expressed with and without SPARQL

Figure 2.5 presents two sample access policies, expressed with and without SPARQL. The policy visualized in Figure 2.5a allows read-only access to the protected resource exclusively by a specific user and from a given location. The policy in Figure 2.5b authorizes the update of the resource by the given user, only if he is currently near Alice. Whenever an HTTP query is performed on a resource, Shi3ld runs the authorization algorithm to check if the policies that protect the resource are satisfied or not. The procedure verifies the matching between the client attributes sent with the query and the access policies that protect the resource. Shi3ld deals with authorization only. Nevertheless, authentication issues cannot be ignored as the trustworthiness of client attributes is critical for a reliable access control framework. Shi3ld supports heterogeneous authentication strategies, since the attributes attached to each client request include heterogeneous data, ranging from user identity to environment information fetched by device sensors (e.g. location). The trustworthiness of user identity is achieved thanks to the WebID6 compatibility: in Shi3ld, user-related attributes are modeled with the foaf vocabulary[7], thus easing the adoption of WebID.

Authenticating the attributes fetched by client sensors is crucial to prevent tampering. Hulsebosch et al. [50] provide a survey of verification techniques, such as heuristics relying on location history and collaborative authenticity checks. A promising approach is mentioned in Kulkarni and Tripathi [58], where client sensors are authenticated beforehand by a trusted party. To date, no tamper-proof strategy is implemented in Shi3ld, and this is left for future work. Moreover, sensible data, such as current location must be handled with a privacy-preserving mechanism. Recent surveys describe strategies to introduce privacy mainly in location-based services [36][56]. Shi3ld adopts an anonymity-based solution [36] and delegates attribute anonymisation to the client side, thus sensitive information is not disclosed to the server. We rely on partially encrypted RDF graphs, as proposed by Giereth [44]. Before building the RDF attribute graph and sending it to the Shi3ld-protected repository, a partial RDF encryption is performed, producing RDF-compliant results, i.e., the encrypted graph is still RDF (we use SHA-1 cryptographic hash function to encrypt RDF literals). On server-side, every time a new policy is added to the system, the same operation is performed on the attributes included in access policies. As long as literals included in access conditions are hashed with the same function used on the client side, the Shi3ld authorization procedure still holds.

[7]http://xmlns.com/foaf/spec/

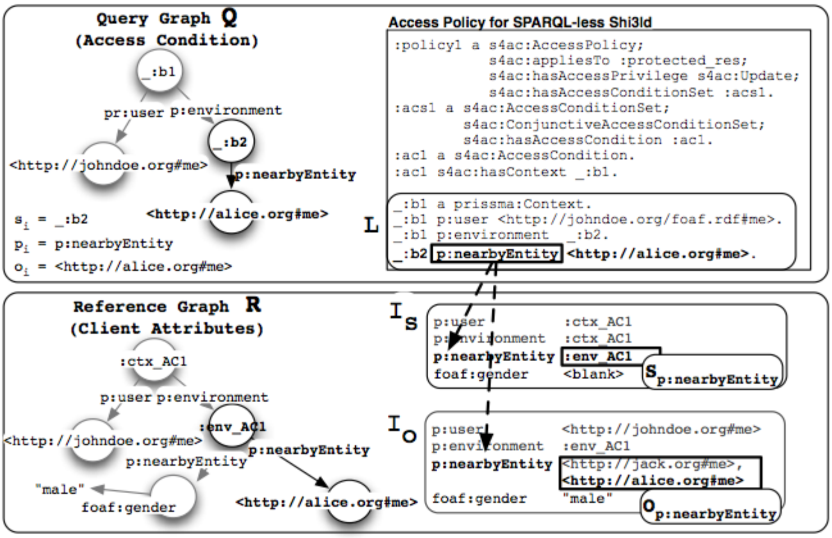

Figure 2.6. Example of subgraph matching used

A SPARQL-less version of Shi3ld

To completely fulfill the Linked Data Platform recommendations, thus achieving a full-fledged basic profile for authorization frameworks, we drop SPARQL from the original Shi3ld implementation. Ditching SPARQL allows RDF-only access policies definition, and a leaner authorization procedure. To obtain a SPARQL-less framework, we re-use the access policy model and the logical steps of the previously described authorization procedure, although conveniently adapted. First, Shi3ld policies adopt RDF only: attribute conditions are expressed now as RDF graphs. Second, the embedded SPARQL engine has been replaced: its task was testing whether client attributes verify the conditions defined in each access policy. This operation boils down to a subgraph matching problem. In other words, we must check if the access conditions (expressed in RDF) are contained into the attribute graph sent along the HTTP client query. Such subgraph matching procedure can be performed without introducing SPARQL in the loop. To steer clear of SPARQL, without re-inventing yet another subgraph matching procedure, we scrap the SPARQL interpreter from the SPARQL engine [31], keeping only the underlying subgraph matching algorithm [8]. To understand how the SPARQL-less policy verification procedure works and comprehend the complexity hidden by the SPARQL layer, we now provide a comprehensive description of the adopted subgraph matching algorithm, along with an overview of the RDF indexes used by the procedure. The algorithm checks whether a query graph Q (the access condition) is contained in the reference graph R (the client attributes sent along the query). The reference graph R is stored in two key-value indexes (see example in Figure 2.6): index Is stores the associations between property types and property subjects, and index Io stores the associations between property types and property objects. Each RDF property type of R is therefore associated to a list of property subjects S_p and a list of property objects O_p. Sp contains URIs or blank nodes, O_p contains URIs, typed literals and blank nodes. Blank nodes are represented as anonymous elements, and their IDs are ignored.

The query graph Q, i.e., the access condition attributes, is serialized in a list L of subject-property-object elements $\{s_i, p_i, o_i\}$[9]. Blank nodes are added to the serialization as anonymous si or o_i elements. The matching algorithm works as follows: for each subject-property-object $\{s_i, p_i, o_i\}$ in L, it looks up the indexes Is and Io using p_i as key. It then retrieves the list of property subjects S_p and the list of property objects O_p associated to p_i . Then, it searches for a subject in S_p matching with s_i , and an object in O_p matching with o_i. If both matches are found, $\{s_i, p_i, o_i\}$ is matched and the procedure moves to the next elements in L. If no match is found in either I_s or I_o, the procedure stops. Subgraph matching is successful if all L items are matched in the R index. Blank nodes act as wildcards: if a blank node is found in $\{s_i, p_i, o_i\}$ as object o_i or subject s_i, and O_p or S_p contains one or more blank nodes, the algorithm continues the matching procedure recursively, backtracking in case of mismatch and therefore testing all possible matchings.

The example in Figure 2.6 shows a matching step of the algorithm, i.e., the successful matching of the triple "_:b2 p:nearbyEntity http://alice.org/me" against the client attributes indexes I_s and I_o. The highlighted triple is successfully matched against the client attributes R. The SPARQL-less version of Shi3ld described in this Section uses the Authorization header to send client attributes. Even if there is no limit to the size of each header value, it is good practice to limit the size of HTTP requests, to minimize latency. Ideally, HTTP requests should not exceed the size of a TCP packet (1500 bytes), but in real world finding requests that exceed 2KB is not uncommon, as a consequence of cookies, browser-set fields

[8]Third-party SPARQL-less Shi3ld-LDP implementations might adopt other off-the- shelf subgraph matching algorithms.

[9]A preliminary step replaces the query graph Q intermediate nodes into blank nodes. Blank nodes substitute SPARQL variables in the matching procedure.

and URL with long query strings[10]. To keep size as small as possible, before base-64 encoding, client attributes are serialized in turtle (less verbose that N-triples and RDF/XML). We plan to test the effectiveness of common lossless compression techniques to reduce the size of client attributes as future work. Furthermore, instead of sending the complete attribute graph along all requests, a server-side caching mechanism would enable the transmission of attribute graph deltas (i.e. only newly updated attributes will be sent to the server). Sending differences of RDF graphs is still an open research topic.

2.2 Semantic Annotation[11]

The main goal of this module is to process the information associated to the content received by the system. Such information (such as the title of the content, user comments or keywords), will be analyzed in order to be linked to the Web of Data. While there are already some interesting works that provide a solution to this problem(see DBpedia Spotlight [63] and the NERD Framework [72]), the decision to create our own semantic annotation module was taken to build an annotator that prioritizes the automatic disambiguation of concepts (since the aim of this framework is to create mobile services mainly targeted to mobile end-users which are rarely interested in giving any feedback to the system).

2.2.1 Automatic Semantic Tagging/Analysis

On the way to semantically cataloging the mobile user-generated content sent by users, the challenging goal of this task is to contextualize the content by analyzing any related information. As previously described, some information is input by the user himself (e.g. title and/or tags) whilst other is automatically derived by the context management module described in 2.1, such as user location or nearby friends. This step aims at associating a semantic LOD (Linked Open Data) concept (and related resource) to every possible set of information. Whilst it may seem easier to link places and people to specific non-ambiguous resources, many challenges arise when trying to identify a ”best resource” that maps to a single user-generated tag or word within a title. Such challenges are still common issues with text mining and natural language processing techniques and this work does not intend to provide a systematic solution to this problem, rather a reasonably fair approach to the main issue of semantically cataloging user-generated content. This solution is described below in details.

[10] https://developers.google.com/speed/docs/best-practices/request

[11] **This section is based on [74]**

Location Analysis

Whenever a content item is received on our system, its sender is identified and, upon success, possibly contextualized. The provided output, whenever available, results in a location (GPS coordinates, civic location and user-labeled place name and type when applicable), and a list of nearby friends (user names and full names). The location information also contains a reference to a Geonames resource, which validity is guaranteed by the positioning process itself based on the geocoordinates. In particular, the (nearest) city-level resource is returned.

Each nearby friend is referenced by a local descriptive RDF resource that also links to external identifiers that the user declared on our platform (such as the user accounts on popular Social Networks). We evaluated the linking of nearby users to existing "external" resources based on the results of a Sindice[12] query: however as the results may be ambiguous and may trigger privacy concerns this further automatic process was turned off and only local linking was retained.

Text Analysis

Title and plain tags analysis follow a complex process to provide candidate resources.

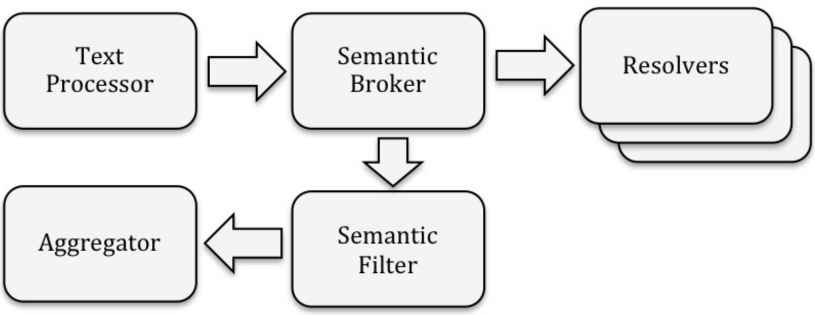

Figure 2.7. Semantic Annotation Process

In particular, semantic resource extraction from title analysis is illustrated in Figure 2.7 and performs as follows: First, a text processing is performed. The title language is initially identified using [8] based on [28]. Then, a morphological analysis is performed using FreeLing [7] configured with the identified language.

[12]http://www.sindice.com

FreeLing was preferred over TreeTagger [14] as it allows for multiwords lemma detection and potentially provides WordNet sense annotation and named entity classification, although not yet on all supported languages (whilst it performs morphological analysis on all languages of our use case). From that analysis, proper nouns lemmas are extracted whilst other part-of-speech are discarded. At this time, non-numeric proper nouns lemmas with a score of at least 0.2 are preserved and merged with plain tags to compute a well-defined list of unique (multi) words. We do understand that nouns or verbs can be useful to describe a peculiar characteristic of the content or the place it was taken (e.g. the subject of a picture or the activity that can be done there) although a further pruning would be required to restrict to concrete concepts only, further discarding abstract statements. At this stage, we thus use term frequency to further process the title and extract other potential relevant words. However, we intend to use the WordNet sense annotation capability of FreeLing for this purpose in the future, coupled with disambiguation techniques and others to select only lemmas.

The next step involves a semantic brokering component assisted by a set of resolvers that perform full-text or term-based analysis based on the previous output. Such resolvers are aimed at providing candidate semantic concepts referring to LOD as well as additional related information if available. Resolvers may be domain or language-specific, or general purpose. For term-based analysis, each word of the previously computed list is individually processed to identify a list of candidate LOD resources to match with. Such activity is accomplished by leveraging techniques provided by SMOB. In particular, we rely on the provided framework for invoking a set of predefined services, such as DBpedia and Sindice, further extended to the Evri[13] entity resolver. In particular, whilst most of such resolvers provide Web service type of interfaces, the DBpedia query has been optimized to rely on SPARQL rather than on its lookup service. This functionality allows to benefit from the full-text support, as well as additional filters e.g. based on language, entity type & native scoring. The query also follows resource redirections to avoid returning "disambiguation" pages (whenever the "wikiPageDisambiguates" property exists). In addition, we further realized that in some cases named entity recognition would benefit from the original context (the whole title) to help disambiguation. As such we also rely on full-text based resolvers such as Evri and Zemanta[14] to derive additional candidates.

Next, a semantic filtering applies to process candidate LOD resources received by the broker and perform disambiguation. Resolvers return many candidate resources that eventually match with the actual concept behind the single word.

[13] http://www.evri.com

[14] http://www.zemanta.com/

Whilst the disambiguation task is humanly solved in the case of semantic search and browsing of content where a dynamic user interface is proposed to the user for selection, our goal is to automatically select and discriminate the most appropriate candidate resource. This disambiguation process starts with a priority and validation mechanism to filter and rank the candidates resources extracted. Such mechanism is based on the analysis of the target resources retrieved and their associated ontologies/graphs. We decided that resources referring to the Geonames graph have higher priority than the ones related to DBpedia, followed by Evri types of resources. At this time all candidate resources pointing to other graphs are discarded. Note that for some resolvers, e.g. Sindice, candidate resources may refer to various ontologies, e.g. Geonames or DBpedia or others. For this reason we associate priorities with graphs and not with the resolvers.

The reason we give Geonames the highest priority is mainly for consistency with our contextualization process in which locations are systematically linked to a Geonames resource. Second, besides being very exhaustive on locations, Geonames is also focused on this types of resources where very little overlap exists with other types of resources. For this reason we prefer it over DBpedia, which remains our preferred graph for generic and heterogeneous concepts, especially in the context of tourism, points-of-interest and celebrities, where rich and well-linked data is available.

Besides this priority mechanism, a validation is performed to check whether the resource itself is valid. This step depends on the provider. For example, for DBpedia resources, and only when the candidate resource is not coming from the DBpedia resolver, which already performs a similar disambiguation check, we query the SPARQL endpoint to check whether they contain an actual binding, or whether they correspond to a disambiguation page. Resources that do not validate are discarded.

In order to provide automatic annotation, a final disambiguation step is needed on the remaining candidates. We implemented this through a function based on the DBPedia score and the string similarity between each surface form and its corresponding list of candidates, based on the Jaro-Winkler distance. This function aims at maximizing both values to identify the "preferred" candidate. In this process, after several empirical tests, candidates with distance lower than 0.8 are discarded at this stage, unless their DBpedia score is maximum. Automatic annotation is performed using the "preferred" candidate identified during this step. Still empirical tests proof that such technique can be further improved to limit false positives: we plan to add a semantic-based validation step in the future, to check consistency between entity types and relations of the overall set of identified resources.

2.3 Content Management

This module not only aims to provide a solution for storing "files", but to offer a number of features that make it indispensable for the storage and retrieval of user-generated content, including its related information and semantic annotations.

As it can be seen from the architecture diagram, this module comprises three submodules that intercommunicate with each other to function properly. In this way, with each content element and its related information can be stored integrity. Thus, we have:

- Data store: comprises a content management system developed specifically to meet the needs of the system, it uses a relational database that stores the information regarding the content, exactly as it is received from other system modules. Likewise assigns a unique identifier to the content that will be used throughout the content lifecycle. This identifier is the same that will be used by the other two submodules described below. Technically, it is a system developed under the LAMP[15] solution stack, in this case with MySQL[16] as the database and PHP[17] as the scripting language.
- Multimedia store: when it comes to a multimedia content, the file content is store in the file system that this module provides. The path where it is located, will be saved by the Data Store submodule for future recovery.
- Triple store: Once the semantic annotations have been extracted from the information of a given content, an RDF containing them will be inserted into this module by the Semantic Annotator (described in 2.2). It should be noted that the RDF has information on the ID number of the content in the Data Store. Thus when queries are made to this module (using the SPARQL endpoint available), the ID number can be easily obtained in order to retrieve the multimedia content itself. Technically speaking this submodule is based on an Openlink Virtuoso Commercial Edition server[18], and has been tested running on a 32bit dual core processor Virtual machine with 12GB RAM.

2.4 Content Retrieval

The content present in the system can be recovered in different ways:

[15] http://en.wikipedia.org/wiki/LAMP

[16] http://www.mysql.com

[17] http://php.net

[18] http://virtuoso.openlinksw.com

- Semantic-based: making more complex SPARQL queries on the endpoint provided by the content management module
- Criteria-based: invoking the APIs provided by the content management module it is possible to retrieve content organized with predefined criteria. For example:
  - Last 10 content elements received
  - Content generated by a given user
  - Content generated in a specific date

Additionally, through this module, the system allows to retrieve content related to a specific element. This is possible through the implementation of a categorization algorithm that discovers relationships between elements in the platform by exploiting the knowledge present in the Web of Data. The algorithm is explained in detail in the next section. This module opens the possibility to service creators to build more complex content recommenders on top of it, or simply to be able to suggest possible interesting content to their users.

2.4.1 Linked Data Concept Recommender

The proposed approach discovers resources that share semantic relationships with a reference resource received as input. To do that the approach relies on two graphs that constitute the knowledge base to semantically discover resources: the resources graph and the categories graph (Linked Data knowledge).

- **Resources graph (G_R):** as the repository of resources is a triple store; resources are RDF entities interlinked by different kinds of relationships among them in the form of a semantic graph namely the *resources graph*. Each node $r_i \in G_R$ is composed of a label lr_i and a description d_i such that $r_i = \{lc_i, d_i\}$, and the edges are the properties (predicate) linking the resources, e.g. $e_{rv}(r_i, r_j)$ is the edge that relates the resource r_i with r_j . In this approach we only consider the generic relation "hasProperty" that represents the predicate of the triples that link the resources stored in the repository.
- **Categories graph (G_C):** this graph is composed of categories extracted from DBpedia and their relationship. In this graph each node $c_m \in G_C$ is composed of the label lc_m such that $c_l = \{lc_m\}$, and edges correspond to the relationships between categories "skos:broader" and "rdf:subClassOf", e.g. $e_{cu}(c_m, c_p)$ the edge that relates the categories c_m and c_p according one of the above mentioned relationships. This graph is specially important because it is the base to infer new relationships between resources taking into account

not only the properties included in the edges of the resources graph but also the categories to which they belong to.

This algorithm discovers resources sharing relationships with a reference resource through the categories graph and the resources graph. Resources are ranked according their semantic distance with the reference resource which is evaluated by counting the number of direct and indirect links between them in a similar way as proposed by Passant [68]. The author defines three functions to calculate the semantic distance $LDSD$ (Linked Data Semantic Distance): the first C_d is the number of direct and distinct links between resources in the resources graph G_R, the second C_{io} and third C_{ii} are the number of indirect and distinct links, both out coming and incoming between resources in the graph G_R. Those functions are integrated in the equation of the $LDSD$ presented in [68].

Our approach considers also the category graph as part of the relationships, inferring in this way new links between resources. Therefore we have modified equation of the $LDSD$ by adding a function C_{ic} which compute the number of links between resources traversing at least one category of the categories graph and a maximum of two categories.

For example, figure 2.8 shows an example of a fragment of the resource graph G_R composed of five resources $(r_1, r_2, r_3, r_4, r_5)$ and five edges five edges with two different types (properties) e_1, e_2. Additionally two nodes of c_1 and c_2 of the categories graph G_C and its edges are presented.

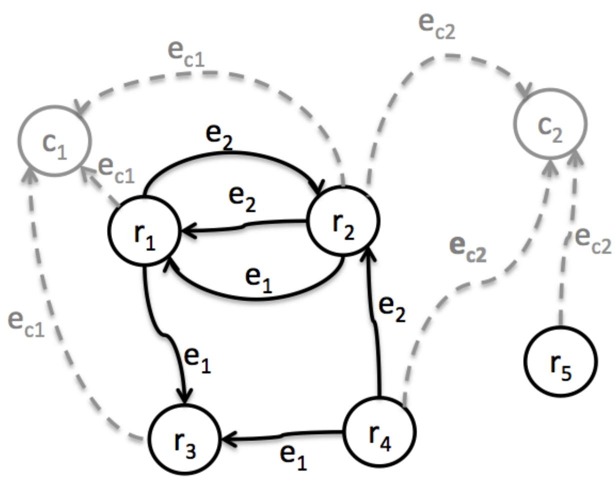

Figure 2.8. Fragment of categories graph (dotted line) and resources graph (continuos line)

Algorithm 1 Resource Discovery Algorithm

Require: Reference Resource (r_{in}), Category Graph (G_C), Resources Graph (G_R)
Ensure: Ranking of first ten resources with the least semantic distance with r_{in}

for all $r_j \in G_R$ **do**
  $Count_d(r_{in}, r_j) = C_d(e_i, r_{in}, r_j) + C_d(e_i, r_j, r_{in})$
  $Count_i(r_{in}, r_j) = C_{io}(e_i, r_{in}, r_j) + C_{ii}(e_i, r_{in}, r_j)$
  $Count_{cat}(r_{in}, r_j) = C_{ic}(e_{ci}, r_{in}, r_j)$
  $LDSD = \frac{1}{1+Count_d+Count_i+Count_{cat}}$
end for
Rank the ten first categories with lower $LDSD$
Assign the ranked categories to the resource r_i

Part III

Mobile Services Built with the Mobile Service Creation Framework

Chapter 3

A Personal Content Sharing Service[1]

3.1 Introduction

The increasing popularity of mobile devices and their continuously growing computing power and location capabilities together with the most innovative Web technologies and services, offers Web users new possibilities to share experiences on-the-go by allowing them to create multimedia content that can be easily distributed and published over the Internet.

On the other hand, as technological improvements are giving a boost to the content creation process, finding suitable content is becoming more problematic for mobile users. Typically, mobile users looking for interesting content related to their current position or POI, access Web search engines from their mobile devices relying on keywords to describe their intent. Unfortunately such descriptions are often subjective and retrieval can be ineffective. In fact, there is no point in making available multimedia information that can only be found by chance [79].

We describe a solution to address the previous issue by enabling user-generated content retrieval through semantic annotations as they give the possibility to quickly access and use the immense knowledge offered by the Web (user-generated or not).

Content published by mobile users is stored in our system and automatically enriched with semantic annotations (using the module described in 2.2) based on the user's context (module described in 2.1) and preferences. Specific interfaces

[1] **This chapter is based on [74]**

for traditional and mobile Web browsers were created to allow semantic-based searching and browsing of stored user-generated content and its related entries on the Linking Open Data cloud and the Web.

3.2 Content Generation

3.2.1 Opportunity Through Semantics

In a mobile scenario, searching and accessing content has to be as relevant as possible. The added value for mobile users is not only to access content itself but also to retrieve the “best content” in a specific situation. If the target is a very effective retrieval process the focus must be, firstly, on how to classify content. For example, on Flickr, image classification is performed by users themselves, thus creating “folksonomies”.

Keyword-based searches, especially when relying on user-generated tags with wild-free vocabulary, restrict the amount of retrievable content and the type of queries that can be performed. However, the main problem of such approach is the ambiguity: the thoughts of a tag creator in a specific situation can be very different of a tag consumer in the same situation. User classification through self-defined tags and tag-based search must trigger the “process of mutual understanding” that involves tag creators and consumers [62].

Trying to overcome this ambiguity when attaching user-generated tags, an extra step is needed: annotate content with semantically meaningful labels referring to an ontology. The creation of a domain ontology is fundamental in the semantic Web context and can be greatly helpful (for example, in a context like tourism, where places, events and people are popular types of entities).

Each uploaded content can hence be stored with these kinds of new and richer tags that allow better management and, accordingly, effective access and retrieval of multimedia data. The annotation mechanism can be distinguished between manual and automatic.

Even though the latter is challenging, in both cases the process must be assisted by a predefined ontology in order to use a common way of annotation in the phases of cataloging and retrieval.

Besides helping in classifying data and content for search, annotation and semantics offer powerful query means that goes beyond tags and keywords. They help to deepen knowledge by allowing creation of structured linked data that can be mashed-up together in several ways based on the query situation.

Keywords and Semantic Annotator, a link to Linked Data

Keywords were one of the critical aspects of the semantization process. Keywords are extremely important as they can be used as an hook to link external resources to our data: if for example, a content item is tagged with the word “Coliseum”, this keyword can be used to retrieve information about “The Roman Colosseum” in the external datasets and use triples to associate that content to this resource.

This idea opens some issues about semantic annotation, namely on disambiguation, as the association between a keyword and a concept is not always as easy as for the “Coliseum” case. This is still an open problem for which we are evaluating different solutions (such as Apache Stanbol[2] or DBPedia Spotlight). However we decided use the module described in 2.2, which performs a basic named entity resolution, language detection and concept retrieval with ranking that we used to perform the LOD linking

3.3 Content Retrieval and Consumption

3.3.1 Semantic-based Virtual Albums

We propose a new semantics-based navigation approach for content retrieval, which provides users with the possibility to reach the best content by exploiting their current situation and context information. Giving users the best content in a specific context means that we have both to provide a strong search instrument and anticipate the users' retrieval needs. That is what our virtual albums do.

A virtual album is a collection of multimedia objects retrieved dynamically by applying several complex search conditions over our data storage. Dynamically organizing multimedia content into virtual albums based on keywords or by simple criteria such as “by user”, “by date” or “by place” was something already possible by means of relational DB technology.

Now, virtual albums have evolved, they're created through pre-generated SPARQL queries made on our Triple Store endpoint and by managing the content rendering: SPARQL is used to express queries across several datasets and its expressiveness helps creating ”complex” queries that are not allowed by the traditional keyword search.

Such queries can be richer, more elaborated and accurate as several matching clauses can be specified to retrieve specific resources and related information, also relying on inference capabilities.

Obviously the user has nothing to do with SPARQL queries, as they are hidden behind web links such as “enrich content”, which will launch the query with parameters taken from the given user's context. In the next example the context gives the system information about the user location, which is “close to Mole Antonelliana”, an important monument in Turin.

[2] http://incubator.apache.org/stanbol/

Select the set of user-generated content, taken near to the monument "Mole Antonelliana"

```
1. PREFIX rdfs:http://www.w3.org/2000/01/rdf-schema#
2. PREFIX sioct:http://rdfs.org/sioc/types#
3. PREFIX comm:http://comm.semanticweb.org/core.owl#
4. PREFIX rev:http://purl.org/stuff/rev#
5. SELECT DISTINCT ?link WHERE {
6. ?monument rdfs:label "Mole Antonelliana"@it .
7. ?monument geo:geometry ?sourceGEO .
8. ?resource geo:geometry ?location .
9. ?resource a sioct:MicroblogPost .
10. ?resource comm:image-data ?link .
11. FILTER(bif:st_intersects(?location, ?sourceGEO, 0.3)) .
12. }
```

To understand how this works, let us start from line 6, in which the query selects a monument resource labeled "Mole Antonelliana" (in case of our triple store, the related DBpedia resource is returned). On line 7, we take the location with the predicate geo:geometry, which is a specific property that has been previously associated by means of Virtuoso platform capabilities[3]. A geometry is a position specification which says that a particular object stands over a particular area.

On line 8 we start querying the user-generated data contained in our store. Line 8 selects all geometry-referenced entries, line 9 filters them by selecting only the ones user-generated and line 10 selects from the RDF descriptor the URL of the real content.

Line 11 is a call to a specific Virtuoso geo-related function, which filters the data by distance from the "Mole Antonelliana" monument.

As anticipated above, behind a virtual album stands a SPARQL query, which is able to retrieve the searched content dynamically with very precise search criteria. However, it is also clear that to exploit at best the possibilities of semantic technologies we must be able to present the virtual albums to the final user in the right way, especially in a mobility context.

[3]More info can be found on: http://docs.openlinksw.com/virtuoso/fn__rdf__geo__fill.html and http://docs.openlinksw.com/virtuoso/rdfsparqlgeospat.html

3.3.2 Web interface for traditional browsers

The platform's Web interface for traditional browsers offers an environment to perform many operations: from personal profile and social features management to content browsing or advanced content editing. It's targeted for modern Web browsers and when it is accessed from a mobile device, redirects the user automatically to the mobile interface (giving also the possibility to switch back to the normal Web interface).

3.3.3 Web interface for Mobile browsers

Our mobile browser Web interface aims to be simple and easy to use for mobile users. When the latter is accessed, it will first try to retrieve the user's geographical position (a user authorization dialog will be prompted) by using the mobile browser's location API. With this information, the possibility of filtering geographically the results can be provided to the user. Then, the interface is presented with a search box, where users can input keywords for retrieving a desired content.

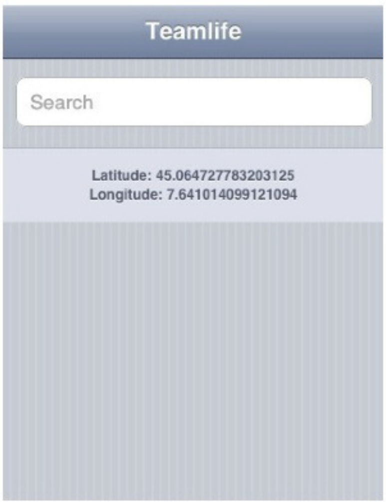

Figure 3.1. Search box and user's location coordinates

The search field is automatic and AJAX-based[4], which means that each time, 2 seconds after the last keystroke is pressed, a query is performed and a list of candidate results will be displayed.

[4]http://en.wikipedia.org/wiki/Ajax_(programming)

Figure 3.2. Result candidates listed for "Turin"

The user can click on the result that matches the semantic sense of his search to visualize all the associated content. It will be presented in a list view with thumbnail previews, brief descriptions and clickable links for a detailed view of each single element.

This view further shows an "About" button on top, which can display a mashup of information related the selected resource based on the LOD datasets.

3.3.4 Enrichment and mashup

As previously said, the semantic query mechanism allows combining data coming from different sources. Data presentation can hence be further enhanced and enriched with linked data coming from the Semantic Web, and is achieved here by clicking the "About" button.

Our main information source is DBpedia, where rich, structured and well-linked data is available. Indeed queries to this source allow collecting interesting and meaningful linked data that is then filtered based on which information and/or content has to be shown to the user.

Starting from a picture sent to our system by the tourist and its semantic location information, useful information is retrieved for the user such as the description (from DBpedia) of the city where the tourist is, the restaurants (and their websites) near the user's location and other touristic attractions in the vicinity that the mobile users can decide to visit, and other UGC content taken in the same

Figure 3.3. Content associated to the selected resource and the "About" button

location from other users.

Figure 3.4. Mobile Interface mashup

3.4 Related Work

Many research activities have been done to provide solutions for information and user-generated content management.

Garcia et al [43], have created a platform which allows the generation, annotation and retrieval of on-the-move multimedia tourist content while on a tourist destination. With a mobile application, a tourist can create content and annotate it based on multilingual folksonomies. An authorized person or body filters it before being published using a web page. This GUI allows browsing for content provided by either individual users or those including a tag or keyword associated to a concept. Generally this work addresses a similar scenario, but the semantic approach is focused only on supporting multilingual annotation through unique concepts, without mentioning semantic meaning extraction or concept disambiguation.

Zhang et al [85], has created a method to demonstrate how Semantic Web can support information integration and facilitate the creation of semantic mashups by integrating the Semantic Web with Web 2.0 in the field of travel planning. We believe the same vision can also be applied to user-generated content, thus, their ideas contributed to inspire our work.

In the area of blogging, Zemanta[5], a content suggestion engine, uses natural language processing and semantic search technology to suggest pictures, tags and links to related articles mainly from social sites. This work inspired us on how to analyze user-generated content automatically.

A very interesting work has been done by Passant et al [69], with SMOB, a microblogging system that provides ways to leverage microblogging according to the Linked Data guidelines, providing semantic annotation tools. This work, together with SparqlPuSH [70] has strongly influenced and inspired our platform semantic enhancements, in particular for the concepts of semantic annotation brokering and of SPARQL-based virtual albums.

3.5 Experimental Results

Currently, our social media platform manages a set of 23772 multimedia contents and 525 users. Afterwards, a set of virtual albums was developed in order to evaluate system performance. Their average rendering time has been estimated in 1.42 seconds.

[5]http://www.zemanta.com

3.6 Conclusions and future work

We presented an approach to introduce and implement semantic capabilities into our existing platform. Mobile users can benefit from automatic semantic enrichment of their content generated on-the-go, which also facilitates the content retrieval process.

As mentioned before, adding automatic analysis capabilities to annotate published content helps us to improve the performance of the content retrieval process and browsing. As the user-assisted disambiguation is not used, it becomes more challenging to guarantee the right semantical meaning extraction, and user evaluations are planned to evaluate and improve our disambiguation algorithms.

Together with the increasing use of contemporary centralized social networks and web applications, the growth of user-generated content over the Web, some issues regarding interoperability and decentralization, privacy and identity and trust building have emerged [38]. Given this scenario, we envision a federation of interconnected social networks and web applications, each one hosted right inside the end-users' home network devices to provide users with full control of their content and information, customization, scalability and interoperability.

At the time of writing, we're planning to move our platform from a centralized to a federated architecture. This step will require so much effort and the need of mature technologies and protocols to make this change possible.

Chapter 4

Social and AR Applications using the User's Context and User Generated Content[1]

4.1 Introduction

The core business of the Mobile Network Operators (MNO) has moved from pure network management and basic traditional phone services in a "walled garden" (such as phone calls and messaging) to a more challenging and not usual area: service providing.

Services in telecommunication, can now be anything from instant messaging between two persons connected to the network, to remote video surveillance by accessing a surveillance station over the Internet.

Potential competition as well as freedom of service creation has significantly increased. The main distinction between an MNO and other Information Communication Technology (ICT) service providers, is that MNOs handle large amounts of their customers' context data (location, presence, availability, etc.) and content (user generated multimedia messages, SMS, video streaming, etc.).

The most important aspect for a business success, is how to identify and use the later specific data to bring value-added services to the customers and therefore, increase the MNO's presence in the market, distinguishing from others and generating solid growing revenues. Telecom Italia, a major Italian MNO, has prototyped

[1] **This chapter is based on [66]**

such type of services after a long research performed in the Context-Awareness and context management field and using its user-generated content management facilities in federation with other platforms and systems. Moreover the operator is continuously involved in many EU projects, leading the Context-Awareness research and integration oriented to a social, augmented-reality, cloud-oriented, value-added service design and creation.

4.2 Technology Enablers

Telecom Italia has been involved in many EU projects aimed to exploit Context-Awareness in different types of application domains and services. The following projects have contributed to the research and development of the Context-Awareness and content management used by Telecom Italia: C-CAST, Mobilife, MUSIC, OPUCE, PERSIST, SPICE, etc.

Currently, Telecom Italia is integrating the Context-Awareness and management platform within the 4CaaSt [1] and FI-WARE EU [3] funded projects where Context-Awareness is embedded into a cloud-enabled service creation and execution environment as a Server Enabler (i.e. Context as a Service Enabler).

The access to this service enabler is performed through a RESTlike interface. In addition, the interface will be extended to support the RESTful OMA defined NGSI standard [9] and enforced with an automatic semantic data tagging and a SPARQL engine to handle semantic requests to the context data. While the technology used by Telecom Italia is pretty mature for a simple provisioning of the context information, its usage is still under an intensive investigation, considering also obstacles posed by privacy laws and regulations in the EU.

The context management framework [65] (described on 2.1.1) is in charge of managing the context data coming from the network and its supporting systems (location, provisioning, etc.), Internet services (social networks, content sharing platforms, etc.) and customers' mobile devices (presence, motion, acceleration, position, speed, location, noise, light, temperature, etc.).

4.2.1 Relevant Standardization

Context, context management and Context-Awareness are already topics of works within following standardization bodies:

- OMA: Mobile Advertising (MobAdv) and Next Generation Service Interface (NGSI) MobSocNet
- IETF: SenML
- Open Geospatial Consortium (OGC)

- W3C: Device API Working Group (DAP)

4.2.2 Supported interfaces

Currently, the interface supported by the Context-Awareness platform is based on HTTP, using a RESTlike interface and/or representation formalisms written in ContextML for simple context information requests and CQL for more complex conditional queries and subscriptions. Moreover, the secondary OMA RESTful NGSI interface is supported as well.

4.2.3 Involved modules

The most relevant modules of the context and content management platforms for the current service development are the following:

- Advanced User Profile (AUP), allows provisioning of the customers' data, including their Social Networks' access credentials (such as username/password or OAuth2 tokens), and a registry of their subscribed services. The applications and the modules described below retrieve user's information from this component.

- Context Broker (CB), it handles the context data (coming from the context providers) that later on can be required by the applications to attach context-based meta-data to user-generated content. It also interacts with the AUP module and applications can invoke it in any moment to retrieve the contextual information of a specific platform user at a given moment in the time.

- TeamLife, a content management system, it is the implementation of the Content Management module described in 2.3. Through its APIs, it is possible to store and retrieve UGC and its associated information. The system is available to the entire platform. By interacting with the Context Broker, content can be automatically enriched with the contextual information available when generated. On the other hand, it communicates with AUP as well, to handle the platform users.

All this modules are also involved into the LOD (Linked Open Data) automatic tagging process of the user-generated content that enables the semantic search and browsing of content, which is out-of-scope of this paper.

4.3 Applications

The most interesting and valuable achievements from a Telecom Italia's perspective are applications involving Context-Awareness and UGC management platforms, which are currently divided into two categories:

- Social Applications
- Augmented Reality

4.3.1 Social eBook reading

Social applications are most relevant for customers with the social appealing.

The book reading, for example, is much more interesting and complete as an experience: if it involves a group of readers, sharing their comments over the same paragraph or even contributing to the book. Groups of readers can be built based on social network relationships. The comments over a reading text can be shared into the social network, thus propagating to other users based on a configuration. Each note can be shared by the users through a specific interface to the most used social network such us Facebook or Twitter. Through this interface friends or followers can see what the users did, read their notes and add comments or retweet the note giving more results to this piece of information. A specific interface from the social network to the system node platform could be provided in order to extract and enrich notes with information provided by users on the social network platforms. One of the most important features is the possibility to use the notes functionality to add automatic information from semantic source. The application provides the current test to a semantic entity extractor platform, the semantic platform recognizes into the text each relevant entity such as places, names, concepts and use it to perform queries to different semantic web sources. The main results could be returned and showed by the application to the user interface giving to the user the possibility to save it as a note that add extra information to the book.

When the system detect and entity as a place, the semantic source could also provide localization information and can be used as extra fields for searching extra contents such as multimedia user generated contents that match the same localization information.

Traditional search could also be provided in order to add extra information using the World Wide Web as a common source for multimedia and extra information, such as image, video, audio and text extra information related to words or sentences written in the book.

At the same time the note platform can provide its information to other applications in order to show note in their target interface based on localization information. Moreover the same application provides some accessibility functions, which extend the book-reading experiences also to the people with limited capacities such as blindness, by reading the text through a text to speech (TTS) engine and low vision, by adjusting the size of the fonts.

This application fits perfectly into education initiatives aiming to schools' digitalization, allowing the interactive education processes to avoid the hard-printed books and making possible the exchange of instant messages (alternative to chat) between teacher and students. The high-level architecture of the service is shown in the Figure 4.1.

At the moment, this service is being exploited by Telecom Italia mainly as a social effect initiative aimed to support limited capacity people and improve the efficiency of the education process in schools. However economic introits are also possible through eBooks distribution supporting this social comments exchange features.

4.4 Augmented Reality

This category of services includes applications that increase the completeness or that customize the perception of the surrounding environment using mobile devices and underlying context-aware content managements systems [82]. Augmented reality leverages two different technologies:

- Sensing by sensors locally embedded into environment, building a smart-space;
- Content enrichment by adding meta-data tagged content available in a remote source, mashed up and linked to a space.

An example of the first service concept is the smart-home concept (currently under evaluation and in trials by Telecom Italia in conjunction with appliance manufacturers), where sensors and smart sensors are embedded into the home appliances. In this case a customer can be updated about their status and maintain the equipment using the real-time online documentation available in place through her/his smartphone. A concept sketch is shown in the Figure 4.2.

The second technology provides additional information to the customer looking at his surroundings through their camera-phone (mobile smartphone equipped with a camera). This information is based on his location, preferences and social relationships. It also provides an augmented view with only the content information relevant for the customers, such the description of a monument, the nearby friends and the details of the near restaurants.

The Augmented Reality content server contains a description of the points of interest (POIs) as geo-tagged data objects and matching the customers' context, including preferences, with the information contained in those objects lead to an

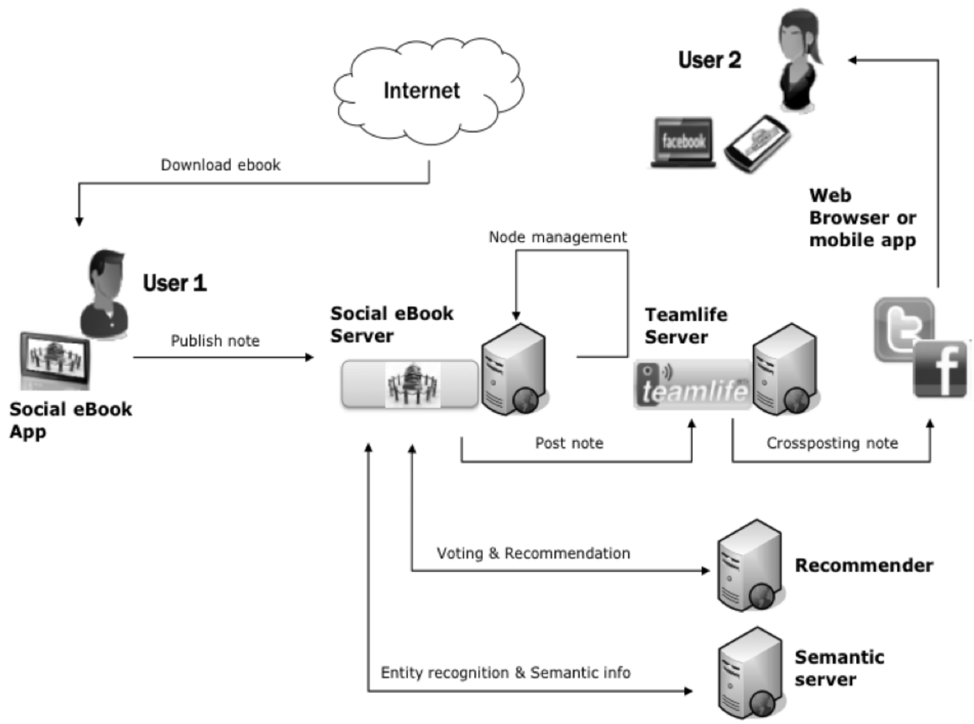

Figure 4.1. Social eBook Platform Architecture

additional visualization layer on the top of the view seen by the phone's camera as shown in the Figure 4.3. Also the customer-related social networks' information such as buddies, friends and friends-of-friends can be visualized near to the customer, thus adding value to the services.

More complex and valuable services can be created by mixing two technologies, hence providing information embedded into the environment (acquired from the local sensors) mixed with information from remote servers, including user-generated content and knowledge about customers' social relationships.

This service is targeted to increase the usage of the data channel provided by the mobile operator and moreover could be either sold as an application pay-per-use or used to increase the appealing of the operator over the competitors.

Figure 4.2. Augmented Reality Application

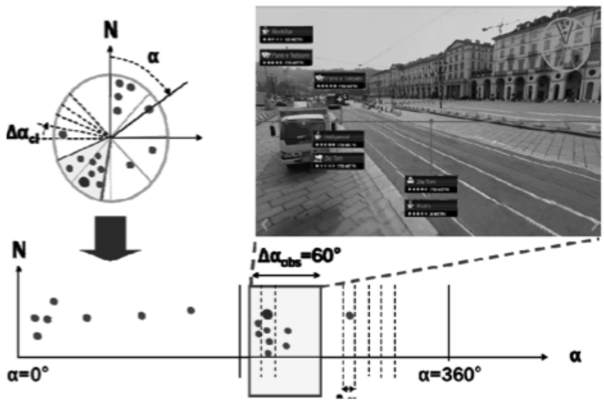

Figure 4.3. Augmented Reality Application View

4.5 Conclusions

The approach of a major Italian MNO to exploit mobile Context-Awareness based services was presented. It goes from building a solid foundation for handling customers' context to the creation of applications on top of it. Innovative applications provide value-added services to end-users by taking advantage of their collected context data and generated content.

Two prototypes were presented: a social reading application, that enhances normal book-reading experiences by introducing the social component and the user's contextual information and an augmented reality application, that brings an enriched perception of the current customer's environment using his location, preferences and social relationships.

By means of its solid platform fundamentals, the operator can focus on the design of new applications to generate novel services to maintain existing clients and also attract new ones, thus generate more revenues. In order to deploy the Context-Awareness platform into a production environment, some issues like user's information privacy have to be acknowledged.

Chapter 5

Standard-Based Publish-Subscribe Service Enabler For Social Applications And Augmented Reality Services[1]

5.1 Introduction

This work demonstrates the adoption of a mature and world-recognized industrial open standard enclosed by an open technological framework, which is implemented as an open reference Service Enabler that can be included and used in a wide range of applications and services involving cross-domain applications.

This reference implementation is based on the open specification of the technological framework defined as an open architecture and, finally, provided as open source code to be used for further integration, adaptation or improvement.

Moreover, it is shown how this technological framework could be easily embedded into a Cloud Computing paradigm. Finally, its reference implementation performed by a Mobile Network Operator (MNO) is demonstrated with real prototyped services as an integrated common technology, potentially available for any

[1] **This chapter is based on [73]**

type of service or application.

The roadmap of this service enabler is given as a future work within the EU funded initiative with conclusions focusing on the relevance of this work for a modern future internet and information enabled Society.

5.2 OMA NGSI Standard For Publish/Subscribe Service Enabler

Future Internet Public Private Partner initiative [2], has been founded by the EU Commission in 2010. It resulted into a number of Use Case Projects (UCP) and a Future Internet Core Platform (FI-WARE project [3]) embracing all the Generic Enablers (GE) commonly used by any UCP. One of the most required GEs identified within FI-WARE is Publish/subscribe GE. The OMA NGSI open standard [9], has been chosen after a careful analysis of the existing industrial open standards and by taking into consideration the already existing solutions provided by the FI-WARE partners. During the selection process, the priority has been given to the practically implemented, existing and used solutions, which are rather simple and able to work with heterogeneous devices through different application domains. OMA NGSI open standard allows to retrieve any type of information (including context data and events), represented as schematically shown in the Figure 5.1, from their respective providers in different modes: on-request and subscription-based.

The information retrieval is performed with the aid of a broker node (as shown in the Figure 5.2).

Additionally, the standard allows the creation of a federation of brokers to avoid bottleneck problems and to yield scalability and flexibility to the final solution. Nevertheless, no specific technological binding has been created within the OMA, which is not aimed at this purpose but perhaps to allow different bindings to be adopted by the interested industries.

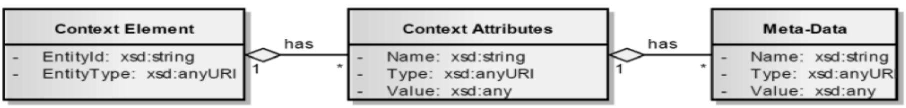

Figure 5.1. OMA NGSI data representation model

Particularly, the FI-WARE project has decided to create a FI-WARE's NGSI RESTful binding based on XML standards and XSD schemas of the data resources parameters and interrogation methods [6]. This decision and the derived technology enable to handle any type of data in a RESTful [13] manner by simply making requests to the data as REST resources. Consequently, the final technological solution will follow the Web service design model [15], which is widely used for the creation of many internet-based services and applications on the web.

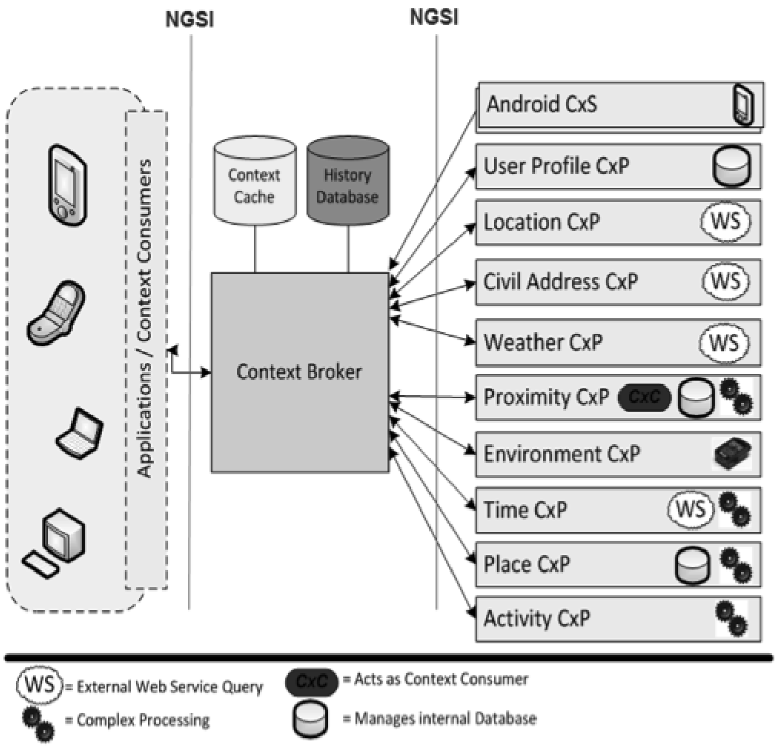

Figure 5.2. NGSI used in the architecture with a Broker

5.3 Open Specification, Architecure, And Open Source Reference Implementation

The FI-WARE project derived the open specification of the publish/subscribe Generic Enabler (GE) [11] based on the Context Management platform (brought into the project by Telecom Italia) and on the OMA NGSI architecture and API specifications. Two sets of interfaces (ContextML/CQL [64] and FI-WARE NGSI [5]) together with the open FI-WARE publish/subscribe GE architecture [12] are published in the FI-WARE web site for public use. Currently, Telecom Italia is developing the reference GE implementation with both types of the interfaces (which also will be available as Open Source): RESTlike ContextML/CQL and Restful FI-WARE NGSI. However, anyone can create its own implementation taking the Open Specifications from the FI-WARE web site.

The main difference between the two interfaces is that the ContextML/CQL implementation (based on the simple exchange of XML-based documents through HTTP requests) has been already in use for long time by Telecom Italia for context management and context-aware applications, thus, it has been proven and is stable. On the other hand, the FI-WARE NGSI interface is still under development (its first release is already available in the FI-WARE project test-bed exposed via FI-WARE GEs Catalogue [4]). It is possible to publish the contextual information of the data producers to the Publish/subscribe broker, which additionally makes it available to be retrieved by context and data consumers (that might be any applications and/or services).

The openness of the overall architecture, the GEs' specifications and their reference implementations available to anyone has a great value for the international and European society as well as it allows anyone to use a world-wide standard based implementation of the Public/Subscribe GE, which is a part of the Future Internet Core Platform. Hence, it is expected that any data generated by context producers will be fully interoperable with context consumers (applications and services) supporting the same GE interfaces[2] on a worldwide arena.

Moreover, due to the plug-and-play architecture of the FI-WARE NGSI allowing to register and remove the context and data producers, the publish/subscribe GE implementation is fully scalable and flexible. Any new data and context providers are able to autonomously connect and register into the publish/subscribe brokers, while the data and context consumers shall not care about where and which data or context information is available.

Finally, comprehensive and extendable query and subscription mechanisms for data and context are supported by both GE interfaces . The subscription allows any data or context consuming entity to be notified by the Publish/subscribe GE when the subscribed data or context is available.

5.4 Publish/Subscribe Enabler Embedded In A Cloud Platform

The Publish/Subscribe GE is based on the broker architecture shown in the Figure 5.2 and it has been embedded into the Cloud platform designed and developed by the 4CaaSt project [1].

[2]FI-WARE NGSI interface will support the query functionality in the Release 2 of the reference implementation. Release 1 available for the moment supports on-request data retrieval mode only

The main goal of the 4CaaSt project is the creation of an automatic platform to allow the deployment and execution of a service or application where publish/subscribe to data and context would be a part of the platform and thus could be used as any other service enabler (data storage, network capabilities, etc.) within the cloud. In order to embed the publish/subscribe and Context-Awareness, the broker designed in the Figure 5.2, has been integrated into the 4CaaSt cloud provisioning, monitoring and charging subsystems. The publish/subscribe and Context-Awareness are now part of the 4CaaSt cloud platform and are implemented as a native service enabler, that is always on and running accordingly to its own dedicated service blueprint deployed an controlled by the 4CaaSt platform.

5.5 Augmented Reality And Social Relevant Applications

In this section, two reference service prototypes developed by a MNO are described in order to show the potential and advantages of the FI-WARE EU project implementation of the OMA's NGSI.

5.5.1 Augmented Reality

In order to improve the daily life of mobile users (users with a mobile smartphone equipped with a camera), an Augmented Reality (AR) service has been prototyped. It brings real-time associated information directly into the objects that the user is watching through the screen of his mobile smartphone (an augmented view). Information is graphically shown through layers, and it is based on mobile user's location, preferences and social relationships.

Some examples of provided information are: monuments' description, nearby buddies, content generated by other users and point-of-interest (POI) details.

A common example application of this service, is the case of a mobile user looking for a place to have a lunch, the AR service can provide her or him with useful information to find the best place considering her or his personal culinary tastes and preferences of the moment. By interacting with the application, the user can access additional information of each recommended place, including information like: the address and phone number of the place, official information related to the point of interest (such as the website), user reviews, expert's reviews or rankings, comments, pictures taken. Additionally, it is possible to "check-in" a place or to see which of the friends of the user have already checked in, as well as the restaurant's menu, price list and ongoing offers and promotions. The architectural components of this service are:

- Augmented Reality Content Server (ARCS), which contains geo-tagged information fetched through different data sources (such as points of interest descriptions or monuments information), mobile users' preferences and social information (such as friends and friends-of-friends). It also handles user-generated content with its related geographical position, for further use of the client application. It is based on the module described in 2.3;
- Client Application, targeted for modern mobile devices. It renders the graphical augmented view by gathering data from the ARCS (based on the geographical position of the mobile user and her or his preferences) and attaching it graphically to the current reality view. It also provides content generation capabilities, it makes possible to upload contextualized user-generated content to the ARCS, which will handle it and manage it for further use. On Figure 5.3, a view from the client application is shown.

This service is mainly targeted to increase the usage of the data channel provided by the mobile operator and moreover could be either sold as an application pay-per-use or used to increase the appealing of the operator over the competitors.

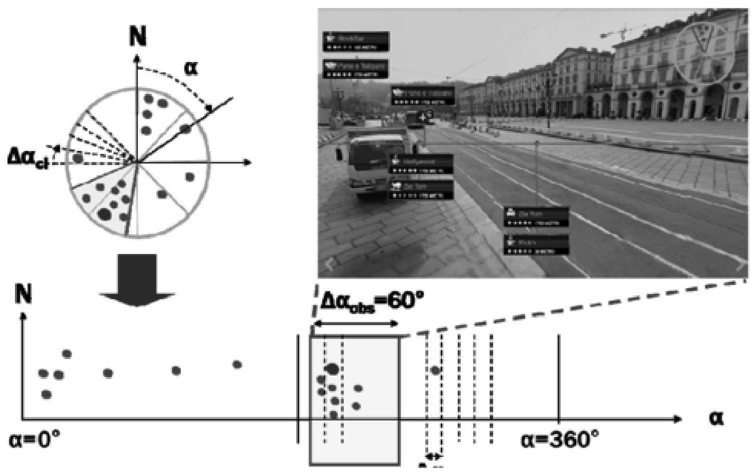

Figure 5.3. Augmented Reality Application

5.5.2 Social Reading

This service has been created with the aim of providing a more interesting and appealing experience of reading books; it focuses on creating a social community around the reader, in particular, those who read electronic books (eBooks) using modern mobile devices. Such community is made up of service users (eBook Readers). Groups of readers can be formed by existing relationships from popular social networks and/or by reading interests.

While reading an eBook through the system's client application, the user has the ability to make annotations or comments about a piece of text or paragraph, which can be also shared with the whole community, with specific groups or selected users (in compliance with the privacy options selected). Sharing with popular social networks like Twitter and Facebook is also possible.

A key feature provided by the service, is the automatic semantic enrichment, it adds associated information from Semantic Web data sources to the notes or comments created by the users. This process is performed by the Semantic entity extractor integrated into the system (which is an implementation of the module described in 2.2), it analyzes them as plain text entries in order to recognize relevant entities such as places, POIs, names and concepts, which are used to perform queries to different Semantic Web sources. The results of these queries are analyzed with a simple algorithm that determines the most probable entity; finally, interesting related content can be shown to the user graphically inside the client application to enrich her or his reading experience.

Non-semantic enrichment is also possible, as the graphical interface of the application also provides options to the reader to attach multimedia content such as images, videos and audio either from web sources or from his existing files on the mobile device.

Strong emphasis has been made on the implementation of accessibility features to bring eBook-reading experiences to people with limited capacities, such as:

- Blindness: The system can aid blind or partially sighted persons by means of its integrated text to speech (TTS) engine. Since the availability of eBooks is greater than audio books, our system provides more reading possibilities. A set of different voices is also available;

- Vision problems: the font and size of the eBooks can be easily adjusted (increased or decreased) and the background color of each page can be selected to the one that fits better;

- Dyslexia: currently, some features being implemented to make the application Dyslexic-friendly.

This service can be used in the education sector, especially to promote the digital education and to avoid the hard-printed books and improving the communication between teachers and students. The high-level view of the service's architecture (shown in the Figure 5.4) is composed by:

- Social eBook App It's a native client application with an interface for eBook reading (at the moment only ePub format is supported) and interacting with the social community; currently, most popular mobile devices are supported;
- Social eBook Server It contains the eBooks available on the platform and at the same time, makes them available for download to the client applications. It manages the notes and comments generated by the users. It is a customized implementation of the Content Management module described in 2.3;
- Sharing Server (Teamlife Server) This serves for communication of the Social eBook Server with the most popular social networks, thus, enabling the "share to" feature in notes and comments;
- Recommender Interrogated by the Social eBook Server and considering the user's preferences, it returns a list of suggested eBook titles;
- Semantic Server Performs the semantic enrichment process of user-generated notes and comments, as described previously.

At the moment, this service is being exploited by a MNO mainly as a social effecting initiative aimed to support limited capacity people and improve the efficiency of the education process in schools. However economic introits are also possible through eBooks distribution supporting this social comments exchange features.

A set of trials of this service has been conducted with a selected group of schools in Italy. The main goal was to collect real usage data and feedbacks to improve the application and its current features.

5.6 Conclusions

In this paper we have described the real-life big effort done by industrial entities to bring their assets for the common usage of the worldwide open community. A Publish/Subscribe GE has been presented as an example of an solution openly defined and based on an open standard demonstrated with a couple of service prototypes created and provided by Telecom Italia. The services are impacting in both the user appealing and the social usefulness perspectives. This effort has been made possible through the support of the European Research Program funding Future Internet Public Private Partnership Program, including the FI-WARE project in which this activity has been performed. Moreover, the results of the EU funded project, 4CaaSt, have been extensively used for the integration of this GE into the cloud technologies and for making it available as a native service for a further usage or service composition and execution.

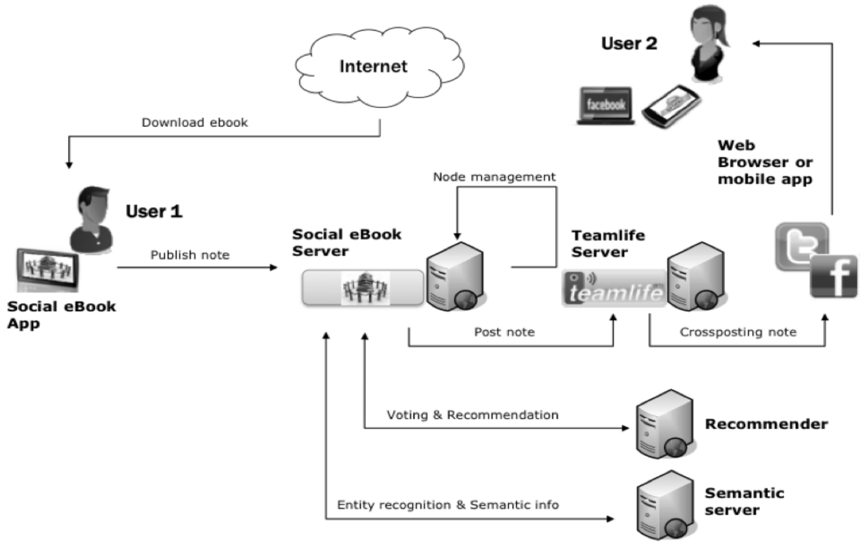

Figure 5.4. Social eBook Platform Architecture

This work is not a final job and there are still a lot of development, implementation and integration activities regarding this GE that will be carried out by the end of 2012 and continuously during year 2013.

5.7 Future Work

The Publish/Subscribe GE issued in its first release already supports the ContextML/CQL RESTlike full mode and FI-WARE NGSI RESTful limited mode communications. This GE will support the full mode of NGSI communications (which does not yet mean a full NGSI support) by the end of 2012 and full NGSI support mode is planned for 2013. However, this implementation, architecture and API specification is already publicly available on the FI-WARE web site. Once the NGSI implementation will be accomplished, most attention will be given to the integration of the Publish/Subscribe GE with the FI Core Platform supporting systems such as monitoring, provisioning and charging capabilities in the FI-WARE Cloud integrating with the work performed within 4CaaSt project. A great attention will be also dedicated to the integration of this GE into the FI-WARE security framework. Finally, starting from the very first release of the Publish/Subscribe

GE by the year 2012, integrations of this GE with other important and relevant GEs within FI-WARE will be performed (such as Big Data, Complex Event Processor, Multimedia Analysis, etc.). At the same time the work of usage and integration of the Publish/Subscribe GE with a number of UCP including OUTSMART, ENVIROFY and SmartCity, has been already started.

Chapter 6

Emergency situations supported by Context-Aware and application streaming technologies[1]

6.1 Introduction

Software systems and new technologies can affect everyone by improving quality of life and can make critical activities safer. These technologies have become one of the most important elements to support many critical situations where people are involved.

During the last years, critical situations have shown the lack of support from telecommunications in case of emergency where different stakeholders are involved. Emergency situations may vary from people in personal difficulties (illness, traffic accident, fire, etc.) to a very broad perspective of social disasters (earthquakes, floods, terrorist attacks, etc.). In such a variety of scenario, effective communication is a very crucial and strategic factor when handling this kind of events. Emergency scenarios usually need very reliable, robust and effective communication systems to enable all the actors to easy coordinate and interact on the field and/or from a control room.

The Next Generation Network (NGN) paradigm is capable to provide advanced

[1] **This chapter is based on [52]**

telecom services addressed to heterogeneous actors who can strongly improve the management of these scenarios, exploiting the capabilities of Internet Technologies while providing new advanced services ([30]) complementary to the traditional ones.

In fact, through an adequate exploitation of the opportunities offered by IT, it is possible to establish communication channels ([51]), especially on large bandwidth networks, which are not limited to voice communication ([16]), but involve the use of ad-hoc applications designed to enhance the flow of information exchanged between the actors. Project PICO aimed at creating innovative telecom services focusing on convergence between fixed-mobile phone networks and technologies for service creation, provisioning, and management, for advanced and high-added-value service delivery. In particular the main project goal was the development and experimentation of an innovative service on large bandwidth networks based on the IMS platform, properly adapted and extended to provide access to both fix and mobile telecom operators.

The flexibility and dynamism of PICO platform can open the way to a new paradigm of managing emergency and crisis scenarios, through a variety of applications devoted and customized to the specific situation depending on the interested scenario. The use of the IMS platform enables to bridge telecommunication and Internet world, opening a huge amount of potential field of applications, which can be used by operators, with high flexibility and ease of access ([29]).

The paper is organized as follows:

- Section 6.2 gives a literature overview of the related work to understand the actual state of the art of different context-aware platforms and approaches focused on specific research fields.
- Section 6.3 introduces the PICO platform.
- Section 6.4 discusses the implementation of an emergency scenario.
- Section 6.5 aims at validating the PICO platform.
- Section 6.6 draws the conclusions and outlines future work.

6.2 Background and Related Work

The emergency concept also covers social telecommunication needs that are mainly related to resources for ensuring public safety; including police, firefighters, ambulance services and other health and medical services, as well as civil defense services. Dedicated networks and equipment for different services meet the telecommunication requirements of such services.

The Project MESA (Mobility for Emergency and Safety Applications, http://www.projectm is an international partnership project between the two major world's standards development organizations, the European Telecommunications Standards Institute (ETSI) and the Telecommunications Industry Association (TIA) of USA. The principal goal of the project is to develop a system for communication between authorities and organizations involved in emergency response or disaster relief activities, maximizing usage of existing communication technologies and infrastructures, even new technologies still under development, and at the same time providing the necessary remedies where current technologies are not able to fulfill users' needs, or when existing infrastructures are not perfectly working and, extremely, are totally down.

Context-awareness ([49]) is a mechanism to present the state information of users, devices, or any entity according to the situation in which they are involved. This mechanism enables Context-Aware platform users to obtain reasonable and proper services through context-aware interactions. [39] describe an approach to represent mobile users' context through an XML-based context representation schema called ContextML. We have been inspired by this work because of its flexibility to represent mobile users' context information into categories and entities, making it also the most powerful solution for our needs.

Service adaptation and Context-Awareness are key points in mobile and ubiquitous computing. Applications on mobile devices use context information to adapt themselves to changing environments. [81] think that user profiles also play an important role in these systems as filter of possible context adaptation parameters. They propose an approach based on the evolution of context-aware user profiles and give a motivating scenario related to the intelligent selection of a suitable medical expert in an emergency situation characterized by user profiles.

On the other hand, many researchers in the mobile communications field also use the Context-Awareness concept to design ubiquitous communicating environments. Nowadays, it is possible to increasingly integrate such services with the public telecommunications by implementing new and reliable technologies. Terrestrial and satellite radio/TV broadcasting, and Internet services provide means for disseminating information in dangerous and critical situations. Currently, operating systems for mobile devices provide different APIs to check the battery state (e.g. battery charge percentage, battery technology and temperature) therefore Software components can be notified whenever the battery-state changes. Many energy-aware applications use certain resources (i.e. GPS, Bluetooth) related to different operations that have a high impact on energy consumption ([24]). Some researches propose adaptation based on the logic and the content. [27] focuses on content adaptation; a typical example of content adaptation is changing the service presentation depending on the context data. The data properties can be modified to adapt the service based on terminal and network capabilities and/or even user preferences. Prior research related to the limited battery lifetime problem is mainly focused on optimizing energy at different levels (e.g. hardware and application layer [35]), including compiler-based energy optimizations, as reported in [41]. On mobile devices, the interfaces that notify the user about device sensors status are not actually enough to provide an adequate evolution of the capabilities of these devices for context reasoning. [57] worked on on predicting user-location based on mobility traces. [42] discuss the importance of Context-Awareness in driving assistance systems. They introduce the concept of collaboration that is pointed out by discussing various situations where cooperation can contribute to make reasonable decisions based on context. They also motivate their work by providing some benefits related to higher comfort of driving and better support for inexperienced drivers in less familiar environments. There is also an interesting work in [78] that presents research on context-aware operational decision support for emergency management. Collaboration and information exchange techniques are useful to provide proper context driving assistance, but there is still a long way until a fully collaborative context sensing system will be available to drivers as a standard feature. Another problem is the lack of more powerful protocols and technologies to share information between vehicles. In [80], authors propose a context-aware middleware-level solution for good environments. Their solution integrates sensors to detect alterations of monitored patients and physical conditions and also detect potentially dangerous situations. Finally, it aims to provide outdoor assistance to the patient. The weak point of middleware context-aware architectures is that they add high computing cost to mobile architectures. The growing population faces different challenges mainly related to high medical costs, low number of doctors and healthcare professionals. The increasing demand for a better healthcare support motivated [71] to propose new solutions based on context-aware collaborative infrastructures for pervasive healthcare which allow an accurate delivery of medical information among the patients, doctors and healthcare workers. Finally, [77] introduce important concepts in the collective ubiquitous context-aware applications field and a methodology for conflict detection and resolution. It is interesting the possibility of implementing a methodology that can maintain the application working properly while solving the collective conflicts.

[20], describe a system that provides communication between victims and rescue teams inside and outside a disaster site by deploying an architecture based on wireless sensor networks (WSN), mobile ad hoc networks (MANET), satellite and cellular gateways. Even if we consider this system a very inspiring work, we believe that in disaster situations is the most valuable resource. When an emergency situation occurs, instead of deploying an infrastructure like the one described by the authors (that can be time-consuming), the use of existing telecommunications systems and infrastructures can save time and in some cases human lives.

Overseer, an open multi-agent system presented in [61], exploits data and contextual information obtained through sensors on mobile devices to provide collaboration and task management for disaster response teams. However, this work does not consider common citizens' context information and behaviors retrieved trough their mobile devices to increase system efficiency.

A system that extends the 3GPP IMS emergency service architecture to provide context-aware enhanced emergency services to 3G users is presented by [37]. One of the framework that can allow government to help people on emergency situations through the use of mobile technologies is described by [18], and it is called m-Government. Nevertheless PICO is not only tailored to work over the existing mobile technologies but it is also designed to exploit the additional capabilities provided by IMS.

In conclusion, the main contributions of our work are:

- A detailed analysis related to the implementation of new technologies in realistic scenarios to provide adequate solutions to handle specific emergency situations,
- A novel context-aware system which integrates different kind of technologies and protocols (communication, streaming, etc.). They provide a context-aware solution able to offer different types of services based on the inferred situation.

According to existing architectures, context-aware approaches are mainly based on collecting and sending useful context data to the reasoning layer, without taking into account new technologies that can be combined to offer more powerful and reliable services. The approaches mentioned before, differ from this work mainly in the aspects related to adaptation according to inferred situations and the integration of IMS technology to provide a more powerful and robust context-aware system compared to other ones.

6.3 PICO platform

This section will show the main feature of the PICO platform. It defines the functional architecture characteristics for gathering, exchanging and providing user context, in order to handle all types of user information and define proper interfaces for interactions over mobile networks.

As shown in figure 6.1, the architecture is composed by:

- Context Manager Client: context acquisition and dissemination,
- Context Manager Server: manipulation, representation, recognizing and reasoning (described in section 6.3.2) about context and situations,

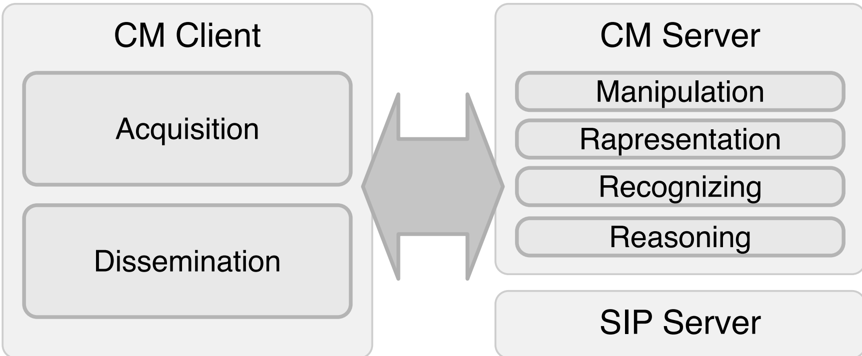

Figure 6.1. The PICO platform architecture.

- SIP Server: video/audio/chat communication and contents transfer.

6.3.1 Context Representation

Definition and representation are fundamental to communicate and process context information. Context-Aware applications need a way to recognize and represent context in order to apply reasoning and adaptation policies ([53]). Context is represented by metadata that is used to describe situations that depend on temporal events or states. The processing and manipulation of this metadata is fundamental to achieve interoperability between actors from different domains and supported by different platforms. A common point in [55] is: context representation (used to implement the reasoning process) requires that the modeling approach should fulfill the next requirements:

- Allow partial validation independently of complex interrelationships.
- Enable rich expressiveness and formalism for shared understanding.
- Indicate richness and quality of information.

Context representation is an essential process in developing Context-Aware systems. Context Meta Language (ContextML) [46] is an XML-based language that enables Context-Aware platforms to discover the user information they need.

The context information is categorized into scopes and related to different types of entities (e.g. user, device). Control messages are a very powerful feature provided by the ContextML framework, because they enable the possibility of adapting different application components.

[39] present an approach for modeling context information and encoding context management messages, they proved that ContextML can be applied for processing and providing high-level context, it can also be used to organize the context in various layers from primitive to high-level scopes.

A strong feature of ContextML scheme is the possibility of adding gradually new scopes and extending the domains of the context. The lightweight representation ensures fast processing performance on mobile devices.

Below we show an example of a ContextML file. ./pictures/asd.xml According to [59], user information is subdivided into scopes, namely, sets related to the same information category. For example, the scope named "position" groups: latitude, longitude, and range regarding a certain entity's location. Scopes can be atomic or aggregated; the latter means the union of different atomic scopes.

Scopes are abstract definitions of a specific set of information and also they are characterized by meaningful names that provide an initial idea about the kind of data contained between the tags.

This data is associated to specific domains of information, and can be included within one or more scopes.

A scope is a simple table of concepts, grouped together and identified by a name, which is used as parameter name in ContextML. The scopes currently defined are location and device information.

These scopes are encapsulated into the ContextML file which is sent from the Context Manager Server to the Context Manager Client.

6.3.2 Context Reasoning Process

Context Reasoning, described in [23], can be defined as the process used to derive logical location and social state from context data, and situation reasoning ([19]) as the identification of the user's activity from context data, logical location and social state. [45] define "situation" as the grouping of three high-level concepts that can easily be used for service personalization.

This process is constantly in execution to:

- Gather raw data;
- Execute different type of operations based on context reasoning.

The main purpose in context reasoning techniques is to include updating context data procedures, according to eventual modifications derived from context information with the scope of increasing the flexibility and improving the adaptability of the system. The main goal of PICO is being able to change its behavior according to a context change.

6.3.3 IMS

The main focus of mobile communications over the past years has been related to bring text, audio and video on mobile devices. Recently, several initiatives have been made to enrich the experience of communication between devices. Mobile communications also introduces the concept of presence information, image and video sharing, instant messaging, and other services. To bring these additional services, IMS framework is commonly used to implement powerful solutions (see Figure 6.2).

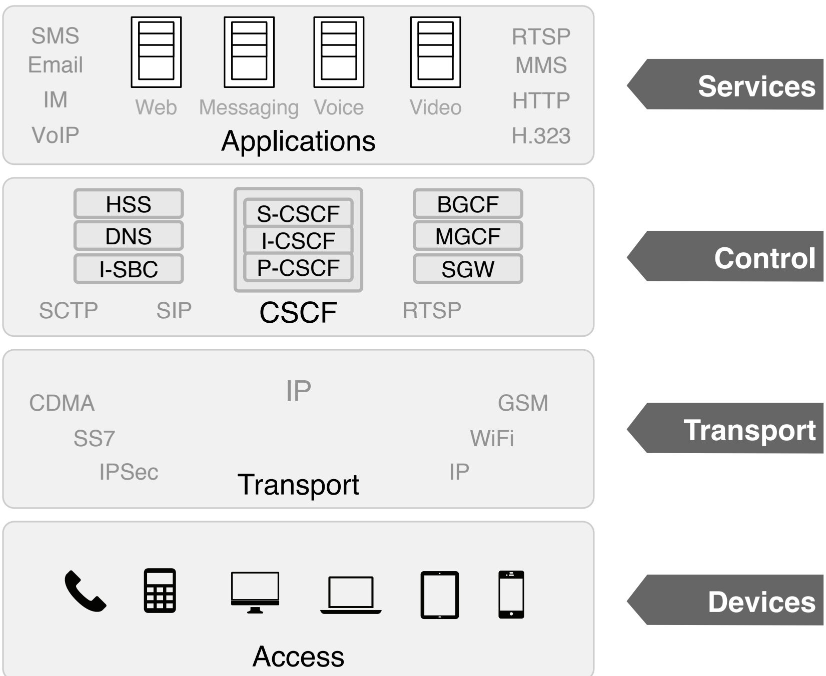

Figure 6.2. IMS architecture

IMS is a framework for delivering IP multimedia services over CS (Circuit Switch) networks (e.g. GSM, LTE). The target is to bring Internet services in mobile networks making interoperable and transparent the circuit switch and packet network. Mainly to satisfy some of these requirements, IMS uses SIP and other protocols such as SIMPLE, XCAP, MSRP, etc.

Before starting to explain the IMS architecture in detail, it is important to have an overview of IMS basic principles that are relevant for the PICO platform:

- Enables access independence. This means that all existing networks could work with IMS, through appropriate gateways and interfaces.
- Allows operators and service providers to use different underlying network architecture.
- Offers extensive IP-based services, such as VOIP (Voice over IP), POC (Push to talk Over Cellular), multiparty gaming, videoconferencing, presence information, instant messaging, content sharing, and so on.

To satisfy the requirements imposed from IMS features, a layered architecture was selected for this architectural framework. From the bottom up, in IMS layered architecture it is possible identifying: Access level, Control level and Applications level. Access level achieves the connection of all users to IMS core network. This is made directly if the actor uses an IMS terminal, and through gateways if the device is not IMS. Gateways use standard interfaces that make possible communicating with all existing entities. This layer is directly responsible for carrying the traffic between endpoints.

Control level has as core task the Call Session Control Function (CSCF). This function is reached by sharing the control between three different entities: Proxy, Serving and Interrogating. Applications level, hosts and executes services (IP applications), and using SIP to interface with the Control layer. IMS Architecture, described previously, is utilized by PICO in the following way:

- User communication device (Context Manager Client): It is a device that operators use as main communication equipment and allows user to interact with the IMS subsystem and its services;
- Application server (Context Manager Server): an application server that belongs to the IMS, and performs specific functionalities such as authentication and authorization of users based on their profiles, context reasoning, mobility handling and application on demand handling.
- Both Context Manager Client and Context Manager Server must be authenticated by IMS, and communication is based on SIP protocol, MSRP protocol, and ContextML files.

As described above, the PICO architecture is based on the IMS Framework. The Context Manager Server performs an IMS registration and it manages the interaction with the Context Manager Clients. It also receives, interprets and analyzes the information sent from the Context Manager Client in contextML format, in order to produce adequate results (e.g., streaming of contents).

All users using a Context Manager Client (extended IMS client) must be registered to the IMS service provided by the SIP server that enables communication (e.g. video/audio call, chat) and content sharing from:

- Context Manager Client to Context Manager Client;
- Context Manager Client to Context Manager Server;
- Context Manager Server to Context Manager Client.

Each Context Manager Client (CMC) is able to contact the Context Manager Server (CMS) by default. The CMC sends periodically its context using the MSRP protocol and the IMS-SIP session.

The context is encapsulated in a XML File (ContextML) which contains several contexts information such as user location, battery level, disk usage, type of user and so on. The Context Manager Server processes the context of all users in a specific area and also analyzes the actual emergency situations; it performs reasoning techniques to send relevant context applications or adaptation orders encapsulated in the contextML file. This file has to be interpreted by the Context Manager Client involved in the emergency situation.

The Context Manager Server (CMS) applies reasoning techniques to perform different actions (e.g., sharing contextML files, applications). The contents are exchanged, organized and installed using MSRP protocol. This protocol extends the IMS platform with full application support. It is relevant considering that application sharing and services deployment procedures need to be performed as fast as possible.

6.3.4 Context Manager Client

Each client is a device, which runs the Android Operating System. The device must have, at least:

- Video camera to perform videoconferencing and allowing recording user sights;
- Microphone to perform audio calls;
- GPS device to perform the geo-location.

A logical view of the device identifies five main blocks (or subsystems) as shown in Figure 6.3:

- Configuration Manager: it provides the configuration of the device in order to communicate with PICO Server
- Authentication Manager: it contains all the credentials required to login the PICO Server

- Context Manager: it manages all the Context changes and sends the information to PICO Server
- Adaptation Manager: it receives all the Context adaptation from PICO Server
- IMS Client: it manages the communication between client and server via IMS

More generally, the IMS client manages all the multimedia sessions required by the Context Manager Client (CMC). It also exports a proper set of APIs to the other Context Manager Client subsystems. Moreover IMS client implements the MSRP protocol to allow file transfer, and more precisely it provides the application transfer from the Context Manager Server.

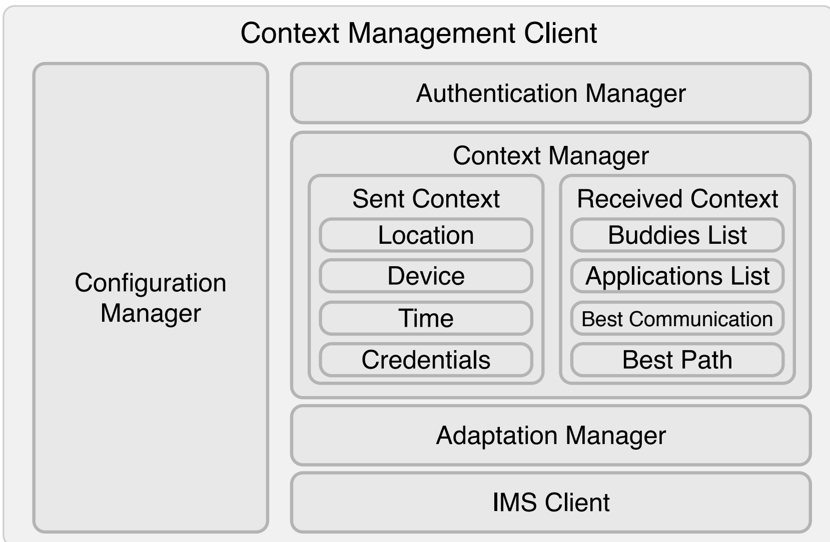

Figure 6.3. Context Manager Client

Once each user performs IMS authentication and gets Context Manager Server authorization, he/she is able to interact with his own interface where it is possible to choose which ”on demand” application it is allowed to download. As said above, the Context Manager Client is an extended IMS client which provides all SIP functionalities such as audio call, video call, chat and file transfer. This communications can be established between users or between Context Manager Client and Context Manager Server. The Context Manager Server (CMS) basing upon a reasoning process decides which Context Manager Clients (CMC) and which services must be connected. The Context Manager Server contains the following elements as shown in Figure 6.4:

- User Profile: all the users have a user profile that describes the configuration for a specific user, including the user's access permission for the ”on demand” applications, user type and preference settings.
- Service Authorizations: this module sets the authorization for the service required at each user level.
- Downloadable Applications: these are the ”on demand” downloadable applications
- Online Applications: ”on demand” applications, which are directly on-line when the user performs the authentication procedure.
- Stream Applications: applications that can be streamed.
- Mobility: this module manages mobility among the three IMS architectures (FF Network, LE Network and EMS Network).
- Core framework: this module performs all the basic functionalities.
- IMS client: performs IMS functionalities and allows registration on IMS, session invite, and file transfer using MSRP protocol.
- Reasoner: performs reasoning based on rules that considers characteristics associated to the kind of emergency, specific area and users location.
- Context: manages all Context parameters based on GEO Coordinates of users, the battery level of the device, network traffic, and other information about the users' context.

6.4 Emergency Scenario

A scenario is the best way to provide a complete overview of the main characteristics involved in emergency situations, which PICO can manage.

This section includes a realistic scenario and a set of screen-shots, which allow the reader to understand how PICO works.

The prototype scenario takes place in Milan. At the beginning there will be no opened emergency in the system. During the course of the scenario a fire emergency near *Piazza Piola* will be activated.

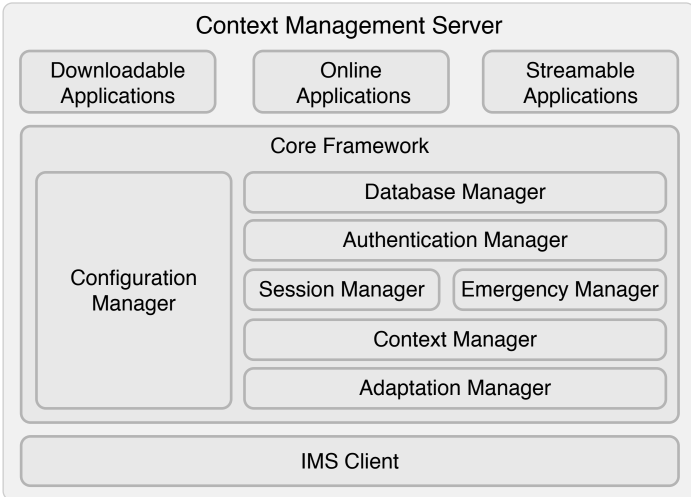

Figure 6.4. Context Manager Server

Figure 6.5 shows an overview of the PICO Architecture and the Actors connected to the platform.

The actors involved for the Prototype Scenario are:

- PICO Server: It is the main application of PICO Architecture. It performs reasoning by analyzing the context of the emergency and the context of the Users connected to the system. It receives all the Users contexts, and it suggests the type of communication to set up between them (Audio/Chat/Video), and provides a set of useful applications to the user.
- Operator: He/She is the operator that corresponds to an agent at the command station. He/She uses an *IMS client* known as *Boghe*. He/She performs an IMS registration to the proxy and is available to receive calls. He can also open an emergency using a web interface.
- Policeman 1: He is the first policeman associated to the emergency. He will show how to open an emergency from the mobile phone.
- Policeman 2: He is the second policeman. He is far from the emergency so he will not be involved in this emergency.
- Firefighter: He is the firefighter associated to the emergency. He will show the application sharing, and how the information read from a QRCode are

propagated to another user connected to the platform.

- Paramedic: He is the paramedic associated to the emergency. He will show how the information sent by the Firefighter are displayed on his mobile device.
- Injured people: in an emergrency there will be some injured people. They have in their wallet a QR code which contains some personal and medical information such as name, number to call, blood group and so on.

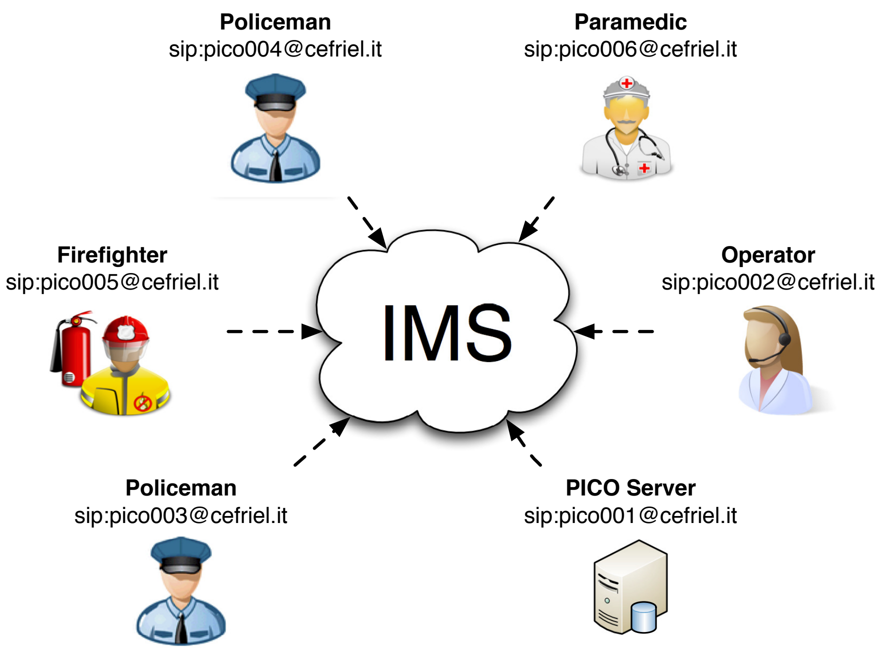

Figure 6.5. Scenario and Actors connected

6.4.1 Scenario Description

Two users (the policemen) activate PICO Mobile application on their devices. The applications will perform an IMS registration and will send the first context updates to the PICO Server. From the Policeman's device, as shown in Figure 6.6a, he will see three main sections. Sections are accessible via a tab navigator which shows:

- Emergency tab. It is composed by (i) an icon which represents the user's type (Policeman, Firefighter or Paramedic); (ii) emergency details; (iii) the buddy list which shows the users connected to the same emergency; and (iv) the map of the emergency.
- Application tab contains the contextual application list proposed by PICO Server

The policeman clicks the operator's icon and open an emergency directly from its device (Figure 6.6b).

After the emergency has been created, PICO Server provides the automatic association between one user and the emergency created. The map shows to the user the best path to reach the location of the emergency (Figure 6.6c).

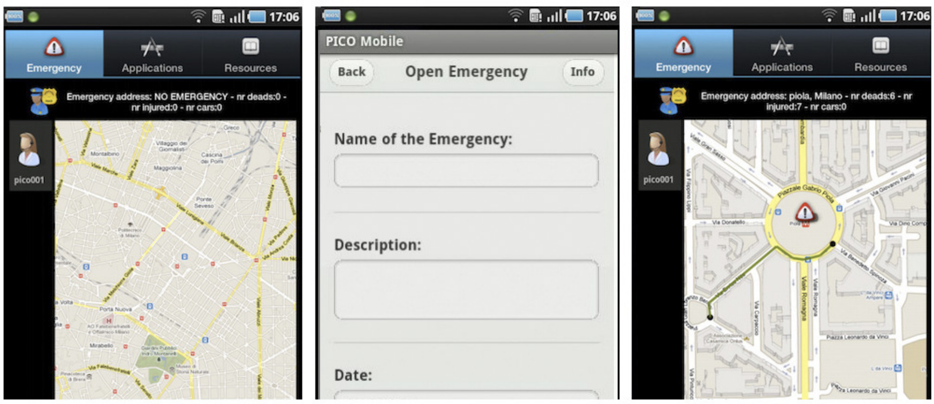

Figure 6.6. a) Starting Screenshot b) Opening an Emergency c) Best Path to Emergency

From the operator perspective, (he/she uses a normal Personal Computer) there is an emergency map which shows all emergencies created and all users associated to.

PICO Server start the reasoning based on the context of the user (location, type, battery level) and the context of the emergency (how many injured, fire, water, outside, inside) and it proposes a list of application pertaining to both contexts. The user can select an application and can start the download of the selected application using the MSRP protocol. This is only an example of what it is possible to do, the server can also send any kind of file or information.

As Figure 6.7a shows, in the application list tab there is a list of applications which can be installed or which has been installed before. Simply touching the

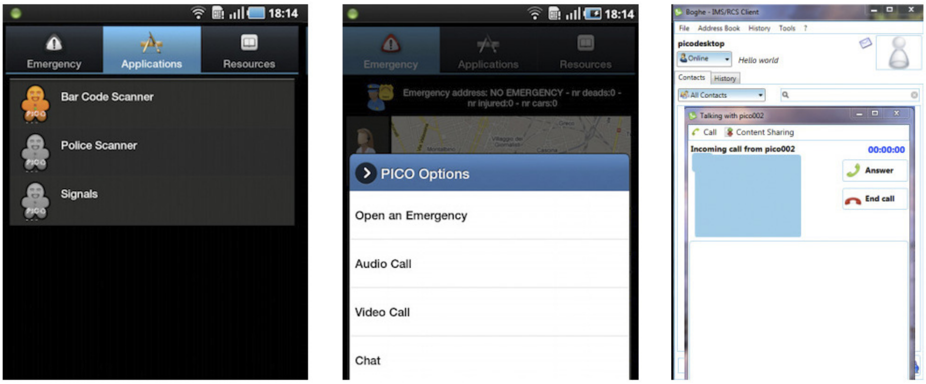

Figure 6.7. a) List of Apps b) Call from Client Side c) Call from Server Side

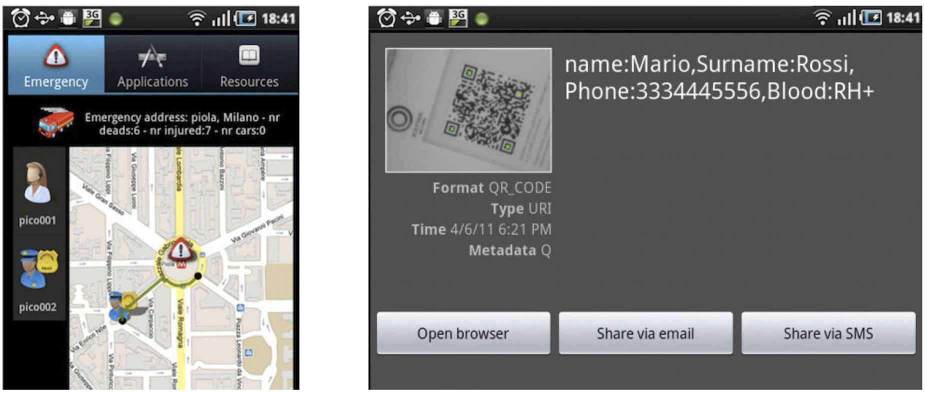

Figure 6.8. a) Firefighter's display b) QRCode reading

application's logo, PICO Mobile will execute it.

By pressing the operator icon, the policeman can perform some operations such as Open an Emergency, call the selected user (Audio,Video,Chat) and send an application. In this case the second policeman will call (audio) the operator to communicate the emergency details (Figure 6.7b).

The operator will receive the audio call from the second policeman using an IMS Client.(Figure 6.7c).

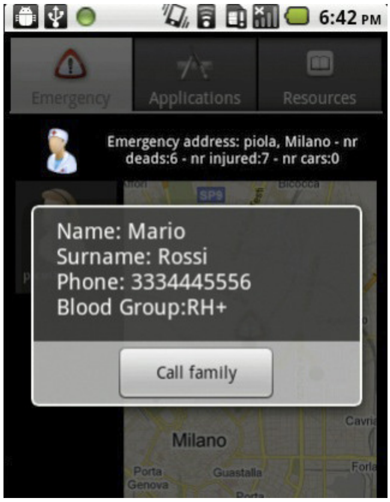

Figure 6.9. QRCode Info from paramedic's device

Figure 6.8a shows the screen of the second mobile user, the firefighter. When he enters into the system, PICO server will automatically associate the firefighter to the emergency. After he decide to use *Barcode Scanner* to obtain useful information about the people injured (Figure 6.8b).

Once the information is read (for the prototype we used a QRCODE, but in the future the NFC could be an alternative), the client will send this information to the PICO Server. The PICO server will send this information over all users associated to the same emergency. This information could help the paramedic (blood group) or other user (call parents of the injured).

As shown in Figure 6.9, the paramedic, who has connected to Pico Platform in the meanwhile, is associated to the emergency in *Piazza Piola*. He will receive all information obtained from the QR Code and he can decide what to do (call the family or communicate the injured's blood group to the hospital).

Time

6.5 Validation and Results

6.5.1 Scenarios

Stress test

Specifically, the scenario for the stress test was the following:

- 3 crises loaded into the PICO Server.
- Each crisis imply 300 users
- The PICO Client Simulator generated 900 users connected PICO Client Simulator producing 4300 context updates, each every 50 milliseconds.

The target is to measure the PICO Server memory usage, the percentage of data loss and the delay of the context updates.

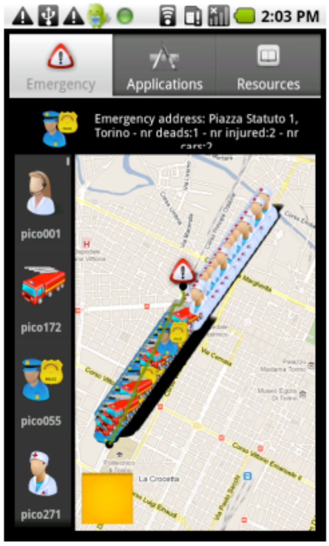

Figure 6.10. Stress test

Endurance test

The scenario for the endurance test was:

- 3 crises loaded into the PICO Server (not close each other).
- Each crisis with 33 users (3 per user type)
- The PICO Client Simulator generated 99 users connected producing 150 context updates, each every 5 seconds.

The target is to measure the PICO Server memory usage, the data loss percentage and the delays of the context updates.

Real emergencies

Composed by:

- 2 types of crises: car accident and fire emergency.
- Each crisis had 3 real users

A 0% data loss and a minimum delay in the communication between server and clients is expected in this test.

6.5.2 Results

Stress test

The memory usage of the PICO Server increased with the number of connected users. When the system had 900 user connected the memory occupation was 85 MB. While from the client side the results registered were:

- 4302 updates sent by the PICO Server
- 179 updates received by PICO Client
- 20 updates per second
- The mean interval between two updates received by the PICO Client: 1208 milliseconds
- 99,04% of data loss

Endurance test

The memory usage of the PICO Server increased with the number of connected users (the number of users connected to the system was directly proportional to the PICO Server memory usage). When the number of users was 99, the maximum memory occupation was 73 MB. While in the client:

- 100 updates sent by PICO Server
- 33 average updates received by the PICO Client
- The mean interval between two updates received by PICO Client was 8695 ms. (see Figure 6.11).
- 67% average data loss

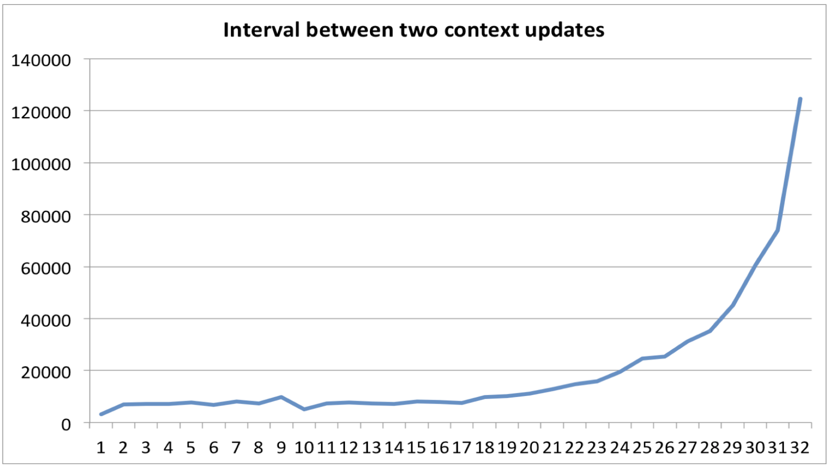

Figure 6.11. Time Interval trend between two context updates

Real emergencies

The maximum memory usage for this scenario was 66 MB.

- 32 updates sent by the PICO Server
- 32 updates received by the PICO Client
- 0% of data loss

6.6 Conclusions

The main goal of this work is to help mobile applications designers to decide what Context-Aware features and streaming characteristics have to be considered when it is necessary a flexible customization of the application behavior to support realistic emergency situations.

Based on what has been implemented, the architecture is feasible and its application appears to be robust. On the other hand we have found some limitations related to size of the files transferred from the Context Manager Server to the IMS client integrated to the Context Manager Client. At the moment small size files are transferred without problem from CMS to CMC.

Evaluating the results of the prototype tests, it was determined that data loss is high because each context update (contextML) implies an update to all other users.

Supposing that the platform has N users and one of them sends an update, N-1 contextML updates are generated. When each user of the platform contemporary sends an update, a total of N*(N-1) messages are generated at the same time. This was confirmed in our tests with 900 users. A possible solution to improve this aspect of the prototype is by implementing a mechanism of selective updates, whether selecting by position (updates are sent only to nearby users) or by setting priorities (allowing to send only the most significant updates).

There are still many relevant factors to improve the prototype efficiency, mainly related to new available features and promising protocols that could be integrated to it. Improving the prototype efficiency is a process that has to consider reduction of the energy consumption ([60]) and proper techniques to make the system architecture more distributed and optimized. Making the architecture more robust from a distributed point of view is a process that has to consider adequate protocols such as LoST (Location to Service Translation) described in [76].

As part of the overall Internet emergency services architecture, LoST, would allow the Context Manager Server to map location data into URLs. LoST is designed to operate globally, with a highly distributed architecture and is an excellent option to fulfill the distributed architecture requirements for emergency situations. This protocol could distribute the Context Manager Server in different nodes according to different kind of rules (e.g., network traffic, server overloading). Such rules are mainly based on limited geographic areas and Context Manager Client requests could be directed to the most appropriate Context Manager Server.

The main result of this work is a reliable context-aware prototype supported by IMS features, which provides relevant information and/or services/applications based on current user context. Providing effective and proper resources (e.g., behavior policies, applications on-demand) to face critical situations is a procedure that depends on how a Context-Aware application is customized according to the current situation and how effective is the platform. The platform is able to handle the interaction between the different actors (e.g., Context Manager Server, Context Manager Client, SIP Server) and it is appropriate to manage emergency situations.

Part IV

Conclusions and Future Work

This dissertation has presented a modular framework to facilitate the process of creating mobile services that take advantage of context information, content generated from their mobile users and knowledge and advantages of the Web of Data to provide better experiences, capabilities and personalization. Likewise, the use-cases of some mobile services targeted to different application domains, such as:

- eTourism
- Augmented reality
- Social reading
- Emergency Situations

have been also described. Each of them, in different ways, due to the inclusion of the aforementioned elements has been able to provide to its users new innovative features.

It should be noted, that the initial idea was that all the modules of the framework presented, should have been implemented in the mobile services discussed. Due to the complexity and heterogeneity of them, it was not possible in all the cases:

- In none of the cases described, the Semantic-based Context Management module has been used. This because the ContextML-based Context Management Module has already proven to be mature and stable for real life services, while on the other hand the technical implementation of the Semantic-based module is currently under testing phase by the same telecommunications operator. As future work, it will be deployed right inside the services described, and also special interest will be put to incorporate the context-based content access control.
- Similarly, any of the described mobile services implemented the Linked Data based Concept Recommender module. This due to the fact that at the time of writing this thesis, a paper that demonstrates its use on a real life mobile service of augmented reality, is being evaluated.

Bibliography

[1] *4caast project web-site*, http://4caast.morfeo-project.org/.

[2] *Fi-ppp*, http://www.fi-ppp.eu/.

[3] *Fi-ware*, http://www.fi-ware.eu/.

[4] *Fi-ware ge catalogue, test-bed web site*, http://catalogue.fi-ware.eu/enablers.

[5] *Fi-ware ngsi open restful api specification*, http://forge.fi-ware.eu/plugins/mediawiki/wiki/fiware/index.php/FI-WARE_NGSI_Open_RESTful_API_Specification_%28PRELIMINARY%29.

[6] *Fi-ware ngsi xsd restful binding*, http://forge.fi-ware.eu/plugins/mediawiki/wiki/fiware/index.php/FI-WARE_NGSI_Open_RESTful_API_Specification_%28PRELIMINARY%29.

[7] *Freeling*, http://www.lsi.upc.edu/~nlp/freeling/.

[8] *Lang detect*, http://pear.php.net/manual/en/package.text.text-languagedetect.php.

[9] *Oma ngsi open specification*, http://www.openmobilealliance.org/Technical/release_program/ngsi_v1_0.aspx.

[10] *Prissma vocabulary*, https://ns.inria.fr/prissma/v2/prissma_v2.html.

[11] *Publish/subscribe broker in fi-ware*, http://forge.fi-ware.eu/plugins/mediawiki/wiki/fiware/index.php/Data/Context_Management#Publish.2FSubscribe_Broker.

[12] *Publish/subscribe ge architecture and open specifications web-site*, http://forge.fi-ware.eu/plugins/mediawiki/wiki/fiware/index.php/FIWARE.ArchitectureDescription.Data.PubSub.

[13] *Rest specification de-facto*, http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm.

[14] *Tree tagger*, http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/.

[15] *Web service design model description*, http://en.wikipedia.org/wiki/Web_service.

[16] Ash Mohammad Abbas and Oivind Kure, *Quality of service in mobile ad hoc*

networks: a survey, Int. J. Ad Hoc Ubiquitous Comput. **6** (2010), no. 2, 75–98.

[17] Gregory D. Abowd, Anind K. Dey, Peter J. Brown, Nigel Davies, Mark Smith, and Pete Steggles, *Towards a better understanding of context and context-awareness*, Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing (London, UK, UK), HUC '99, Springer-Verlag, 1999, pp. 304–307.

[18] K Amailef and Jie Lu, *m-Government: A framework of mobile-based emergency response systems*, Intelligent System and Knowledge Engineering, 2008. ISKE 2008. 3rd International Conference on, 2008, pp. 1398–1403.

[19] C.B. Anagnostopoulos, Y. Ntarladimas, and S. Hadjiefthymiades, *Reasoning about situation similarity*, Intelligent Systems, 2006 3rd International IEEE Conference on, sept. 2006, pp. 109 –114.

[20] Yong Bai, Wencai Du, Zhengxin Ma, Chong Shen, Youling Zhou, and Baodan Chen, *Emergency communication system by heterogeneous wireless networking*, Wireless Communications, Networking and Information Security (WCNIS), 2010 IEEE International Conference on, 2010, pp. 488–492.

[21] C. Baladron, J. Aguiar, B. Carro, and A. Sanchez-Esguevillas, *Integrating user-generated content and pervasive communications*, Pervasive Computing, IEEE **7** (2008), no. 4, 58–61.

[22] Matthias Baldauf, Schahram Dustdar, and Florian Rosenberg, *A survey on context-aware systems*, Int. J. Ad Hoc Ubiquitous Comput. **2** (2007), no. 4, 263–277.

[23] B. Beamon and M. Kumar, *Hycore: Towards a generalized hierarchical hybrid context reasoning engine*, Pervasive Computing and Communications Workshops (PERCOM Workshops), 2010 8th IEEE International Conference on, 29 2010-april 2 2010, pp. 30 –36.

[24] Mejia Bernal, F Jose, Luca Ardito, Maurizio Morisio, and Paolo Falcarin, *Towards an efficient context-aware system: Problems and suggestions to reduce energy consumption in mobile devices*, 2010 Ninth International Conference on Mobile Business and 2010 Ninth Global Mobility Roundtable ICMBGMR (2010), 510–514.

[25] C Bizer, *The Emerging Web of Linked Data*, Intelligent Systems, IEEE **24** (2009), no. 5, 87–92.

[26] Christian Bizer, Tom Heath, and Tim Berners-Lee, *Linked data - the story so far*, Int. J. Semantic Web Inf. Syst. **5** (2009), no. 3, 1–22.

[27] László Böszörményi, Hermann Hellwagner, Harald Kosch, Mulugeta Libsie, and Stefan Podlipnig, *Metadata driven adaptation in the admits project*, Sig. Proc.: Image Comm. **18** (2003), no. 8, 749–766.

[28] William B. Cavnar and John M. Trenkle, *N-gram-based text categorization*, Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval (Las Vegas, US), 1994, pp. 161–175.

[29] Wei-Kuo Chiang and Po-Chiang Kuo, *Decentralised secure handover in ims-based networks*, Int. J. Ad Hoc Ubiquitous Comput. **10** (2012), no. 3, 137–148.

[30] Chi-Huang Chiu, Hsien-Tang Lin, and Shyan-Ming Yuan, *Cloudedge: a content delivery system for storage service in cloud environment*, Int. J. Ad Hoc Ubiquitous Comput. **6** (2010), no. 4, 252–262.

[31] O. Corby and C. Faron Zucker, *The kgram abstract machine for knowledge graph querying*, Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on, vol. 1, Aug 2010, pp. 338–341.

[32] L. Costabello, S. Villata, O. R. Rocha, and F. Gandon, *Access control for http operations on linked data*, 2013, pp. 185–199.

[33] Luca Costabello, Serena Villata, Nicolas Delaforge, and Fabien Gandon, *Shi3ld: An access control framework for the mobile web of data*, Proceedings of the 23rd ACM Conference on Hypertext and Social Media (New York, NY, USA), HT '12, ACM, 2012, pp. 311–312.

[34] Anind K. Dey, *Understanding and using context*, Personal and Ubiquitous Computing **5** (2001), 4–7.

[35] Fred Douglis, P. Krishnan, and Brian N. Bershad, *Adaptive disk spin-down policies for mobile computers*, Proceedings of the 2nd Symposium on Mobile and Location-Independent Computing (Berkeley, CA, USA), USENIX Association, 1995, pp. 121–137.

[36] Matt Duckham, *Moving forward: Location privacy and location awareness*, Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS (New York, NY, USA), SPRINGL '10, ACM, 2010, pp. 1–3.

[37] M El Barachi, A Kadiwal, R Glitho, F Khendek, and R Dssouli, *An Architecture for the Provision of Context-Aware Emergency Services in the IP Multimedia Subsystem*, Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE, 2008, pp. 2784–2788.

[38] K.A. Falahi, Y. Atif, and S. Elnaffar, *Social networks: Challenges and new opportunities*, Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on Int'l Conference on Cyber, Physical and Social Computing (CPSCom), dec. 2010, pp. 804 –808.

[39] Paolo Falcarin, Massimo Valla, Jian Yu, Carlo Alberto Licciardi, Cristina Frà, and Luca Lamorte, *Context data management: an architectural framework for context-aware services*, Service Oriented Computing and Applications **7** (2013), no. 2, 151–168.

[40] Roy Thomas Fielding, *REST: architectural styles and the design of network-based software architectures*, Doctoral dissertation, University of California, Irvine, 2000.

[41] Jason Flinn and M. Satyanarayanan, *Energy-aware adaptation for mobile applications*, SOSP '99: Proceedings of the seventeenth ACM symposium on

Operating systems principles (New York, NY, USA), ACM, 1999, pp. 48–63.

[42] Simone Fuchs, S Rass, B Lamprecht, and K Kyamakya, *Context-awareness and collaborative driving for intelligent vehicles and smart roads*, 1st Int'l Workshop ITS Ubiquitous ROADS (2000), 1–6.

[43] A Garcia, M T Linaza, I Arbulu, I Torre, and Y Cobos, *A mobile platform for on-the-move user-generated tourist contents*, pp. 58–65, 2008.

[44] Mark Giereth, *On partial encryption of rdf-graphs*, International Semantic Web Conference, volume 3729 of Lecture, Springer, 2005, pp. 308–322.

[45] L.-W. Goix, M. Valla, L. Cerami, and P. Falcarin, *Situation inference for mobile users: A rule based approach*, Mobile Data Management, 2007 International Conference on, may 2007, pp. 299 –303.

[46] Laurent-Walter Goix, Luca Lamorte, Paolo Falcarin, Carlos Baladron, Jian Yu, Isabel Ordas, AlvaroMartinez ReoL, Ruben Trapero, JoseM del Alamo, Michele Stecca, and Massimo Maresca, *Leveraging context-awareness for personalization*, pp. 301–334, Chapman and Hall/CRC, 2014/02/16 2010.

[47] Li Han, S. Jyri, Jian Ma, and Kuifei Yu, *Research on context-aware mobile computing*, Advanced Information Networking and Applications - Workshops, 2008. AINAW 2008. 22nd International Conference on, March 2008, pp. 24–30.

[48] Tom Heath and Christian Bizer, *Linked data: Evolving the web into a global data space*, 1st ed., Morgan & Claypool, 2011.

[49] Jenq-Muh Hsu, Wei-Bin Lain, and Jui-Chih Liang, *A context-aware push-to-talk service.*, MUE, IEEE Computer Society, 2008, pp. 586–591.

[50] R. J. Hulsebosch, A. H. Salden, M. S. Bargh, P. W. G. Ebben, and J. Reitsma, *Context sensitive access control*, Proceedings of the Tenth ACM Symposium on Access Control Models and Technologies (New York, NY, USA), SACMAT '05, ACM, 2005, pp. 111–119.

[51] Sheetal Jadhav, Haibo Zhang, and Zhiyi Huang, *Performance evaluation of quality of voip in wimax and umts*, Parallel and Distributed Computing Applications and Technologies, International Conference on **0** (2011), 375–380.

[52] Mejia Bernal J.F., Falcarin P., Ardito L., Rodriguez Rocha O., Morisio M., Pistillo F., and Giovanelli F., *Emergency situations supported by context-aware and application streaming technologies*, 2013, pp. 120–131.

[53] C. Klein, R. Schmid, C. Leuxner, W. Sitou, and B. Spanfelner, *A survey of context adaptation in autonomic computing*, Autonomic and Autonomous Systems, 2008. ICAS 2008. Fourth International Conference on, march 2008, pp. 106 –111.

[54] M. Knappmeyer, S.L. Kiani, C. Fra, B. Moltchanov, and N. Baker, *Contextml: A light-weight context representation and context management schema*, Wireless Pervasive Computing (ISWPC), 2010 5th IEEE International Symposium on, May 2010, pp. 367–372.

[55] Michael Knappmeyer, Nigel Baker, Saad Liaquat, and Ralf Tönjes, *A context provisioning framework to support pervasive and ubiquitous applications*,

Proceedings of the 4th European conference on Smart sensing and context (Berlin, Heidelberg), EuroSSC'09, Springer-Verlag, 2009, pp. 93–106.

[56] John Krumm, *A survey of computational location privacy*, Personal Ubiquitous Comput. **13** (2009), no. 6, 391–399.

[57] John Krumm and Eric Horvitz, *Predestination: Inferring destinations from partial trajectories.*, Ubicomp (Paul Dourish and Adrian Friday, eds.), Lecture Notes in Computer Science, vol. 4206, Springer, 2006, pp. 243–260.

[58] Devdatta Kulkarni and Anand Tripathi, *Context-aware role-based access control in pervasive computing systems*, Proceedings of the 13th ACM Symposium on Access Control Models and Technologies (New York, NY, USA), SACMAT '08, ACM, 2008, pp. 113–122.

[59] Luca Lamorte and Claudio Venezia, *Smart space a new dimension of context.*, PERSIST Workshop on Intelligent Pervasice Environments, april 2009, pp. 1–6.

[60] J. R Lee and D. H Cho, *Dual power-saving modes for voice over ip traffic supporting voice activity detection*, Communications, IET **3** (2009), no. 7, 1239–1249.

[61] F Luqman and M Griss, *Overseer: A Mobile Context-Aware Collaboration and Task Management System for Disaster Response*, Creating Connecting and Collaborating through Computing (C5), 2010 Eighth International Conference on, 2010, pp. 76–82.

[62] Marinchev, *Practical semantic web – tagging and tag clouds*, Cybernetics and Information Technologies **3** (2006).

[63] Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer, *Dbpedia spotlight: Shedding light on the web of documents*, Proceedings of the 7th International Conference on Semantic Systems (New York, NY, USA), I-Semantics '11, ACM, 2011, pp. 1–8.

[64] B. Moltchanov, *Context representation formalism and its integration into context as a service in clouds*, Kaleidoscope 2011: The Fully Networked Human? - Innovations for Future Networks and Services (K-2011), Proceedings of ITU, Dec 2011, pp. 1–8.

[65] Boris Moltchanov, Cristina Fra', Massimo Valla, and Carlo Alberto Licciardi, *Context management framework and context representation for mno*, 2011.

[66] Boris Moltchanov, Carlo Licciardi, Fabio Mondin, Maurizio Belluati, and Oscar Rodriguez Rocha, *Social and ar applications uusing the user's context and user generated content*, 2012.

[67] FALCARIN P., OSCAR RODRIGUEZ ROCHA, LUCA COSTABELLO, and LAURENT WALTER GOIX, *Leveraging context for social media mobile services (in ieee pervasive computing - work in progress section)*, 2008, pp. 58–61.

[68] Alexandre Passant, *Measuring semantic distance on linking data and using it for resources recommendations.*, AAAI Spring Symposium: Linked Data Meets Artificial Intelligence, AAAI, 2010.

[69] Alexandre Passant, Tuukka Hastrup, Uldis Bojars, and John Breslin, *Microblogging: A semantic and distributed approach*, Proceedings of the 4th Workshop on Scripting for the Semantic Web, Tenerife, Spain, June 02, 2008, CEUR Workshop Proceedings, 2008.

[70] Alexandre Passant and Pablo Mendes, *sparqlPuSH: Proactive notification of data updates in RDF stores using PubSubHubbub*, CEUR Workshop Proceedings ISSN 1613-0073, vol. 699, February 2010.

[71] Hung Keng Pung, Tao Gu, Wenwei Xue, Paulito P. Palmes, Jian Zhu, Wen Long Ng, Chee Weng Tang, and Nguyen Hoang Chung, *Context-aware middleware for pervasive elderly homecare*, IEEE J.Sel. A. Commun. **27** (2009), 510–524.

[72] Giuseppe Rizzo and Raphaël Troncy, *Nerd: A framework for unifying named entity recognition and disambiguation extraction tools*, Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics (Stroudsburg, PA, USA), EACL '12, Association for Computational Linguistics, 2012, pp. 73–76.

[73] O. R. Rocha and B. Moltchanov, *Standard-based publish-subscribe service enabler for social applications and augmented reality services*, 2013, pp. 1–6.

[74] Oscar Rodríguez Rocha, Carmen Criminisi, Fabio Mondin, and Laurent-Walter Goix, *Lodifying personal content sharing*, Proceedings of the 2012 Joint EDBT/ICDT Workshops (New York, NY, USA), EDBT-ICDT '12, ACM, 2012, pp. 124–130.

[75] Daniel Salber, Anind K. Dey, and Gregory D. Abowd, *The context toolkit: Aiding the development of context-enabled applications*, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (New York, NY, USA), CHI '99, ACM, 1999, pp. 434–441.

[76] H. Schulzrinne, H. Tschofenig, A. Newton, and T. Hardie, *Lost: A protocol for mapping geographic locations to public safety answering points*, Performance, Computing, and Communications Conference, 2007. IPCCC 2007. IEEE Internationa, april 2007, pp. 606 –611.

[77] Thais R. M. Braga Silva, Linnyer Beatrys Ruiz, and Antonio Alfredo Ferreira Loureiro, *Solving collective conflicts in a disaster emergency attendance application*, ISCC, 2009, pp. 889–894.

[78] Alexander V. Smirnov, Mikhail Pashkin, Nikolai Chilov, and Tatiana Levashova, *Operational decision support: Context-based approach and technological framework*, CONTEXT, 2005, pp. 476–489.

[79] Giorgos Stamou and Stefanos Kollias (eds.), *Multimedia content and semantic web: Methods, standards and tools*, John Wiley & Sons Inc., Chichester, West Sussex, Hoboken, NJ, 2005.

[80] Tarik Taleb, Zubair Md. Fadlullah, Dario Bottazzi, Nidal Nasser, and Yunfeng Chen, *A context-aware middleware-level solution towards a ubiquitous healthcare system*, Proceedings of the 2009 IEEE International Conference on

Wireless and Mobile Computing, Networking and Communications (Washington, DC, USA), WIMOB '09, IEEE Computer Society, 2009, pp. 61–66.

[81] Jan Thomsen, Yves Vanrompay, and Yolande Berbers, *Evolution of context-aware user profiles.*, ICUMT, IEEE, 2009, pp. 1–6.

[82] C. Venezia and M. Marengo, *Context awareness aims at novel fruition models: Augmented reality may be the killer application for context awareness*, Embedded and Ubiquitous Computing (EUC), 2010 IEEE/IFIP 8th International Conference on, Dec 2010, pp. 392–396.

[83] Graham Vickery and Sacha Wunsch-Vincent, *Participative web and user-created content: Web 2.0 wikis and social networking*, Organization for Economic Cooperation and Development (OECD), Paris, France, France, 2007.

[84] Daqiang Zhang, Hongyu Huang, Chin-Feng Lai, Xuedong Liang, Qin Zou, and Minyi Guo, *Survey on context-awareness in ubiquitous media*, Multimedia Tools and Applications (2011), 1–33, 10.1007/s11042-011-0940-9.

[85] Libin Zhang, Qianyu Zhu, Wei Wang, and Pengcheng Zhang, *Web 2.0 and Semantic Web Based Intelligent Service System for Tourism*, Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on (2008), 1–4.