Hybrid context modeling: A location-based scheme using ontologies

Ioanna Roussaki[1], Maria Strimpakou[1], Carsten Pils[2], Nikos Kalatzis[1], Miltos Anagnostou[1]
[1] *School of Electrical & Computer Engineering, National Technical University of Athens, Greece*
{nanario,mstrim,nikosk,miltos}@telecom.ntua.gr
[2] *Telecommunications Software & Systems Group, Waterford Institute of Technology, Ireland*
cpils@tssg.org

Abstract

Context awareness is an inherent feature of pervasive computing. It enhances the proactiveness of the system thus requiring less user attention and fewer human-machine interactions, it supports intelligent personalization features, and it can assist the system to address the user requirements considering the current conditions. Nevertheless, in such environments, various types of context information are involved and need to be efficiently managed and maintained, soundly interpreted, rapidly processed, and securely disseminated by the system. Thus, an interoperable and flexible context representation scheme is necessary that will support efficient context interpretation and reasoning and will perform well in distributed large-scale context-aware systems. This paper is concerned with the development of a hybrid context representation scheme[1] that aims to combine the maintenance, distribution and administrative facilities of a location-based context model and the semantic advantages of context ontologies.

1. Introduction

Pervasive computing [1] will dominate the future of service provisioning systems, where devices, networks and applications will all be seamlessly integrated and will cooperate in support of a worldwide common shared computing paradigm for a plethora of new advanced services. The diffusion of intelligence in everyday objects, due to the miniaturization and cost reduction of hardware, coupled with the research results carried out by ubiquitous and pervasive computing initiatives will change the way providers and customers interact. Eventually, as traditional systems evolve into pervasive, an important aspect that needs to be pursued is context awareness [2].

Context awareness (CA) has the potential to greatly reduce the human attention and interaction bottlenecks, to give the user the impression that services fade into the background, and to support intelligent personalization and adaptability features. There are several research teams that work on the CA area and have developed various albeit similar infrastructures, most of them managing context information in specific application domains and in an ad-hoc manner [3]. In this perspective, context is still a poorly utilized source of information. In an effort to change this, we broaden our research on various, complex topics, an indispensable of which is context modeling.

In the past a variety of context models were studied. Early models mainly focused on addressing the needs of a single application in a simple and straightforward way, while no attempt was made to design and develop generic context models. Later approaches proposed context modeling concepts not tied to specific application domains. However, they usually represent a limited amount of context information. In this framework, research on designing a common context representation scheme is mostly concerned with location, identity, and time. Nevertheless, recently several research initiatives work towards the development of various context modelling techniques [4], representation and query languages, as well as reasoning algorithms that facilitate context sharing and interoperability of applications. Evaluation of recent research work on context modeling [5] indicates that the usage of ontologies exhibits prominent advantages in integrated pervasive environments.

[1]This work has been partially supported by the Integrated Project DAIDALOS ("Designing Advanced network Interfaces for the Delivery and Administration of Location independent, Optimised personal Services"), which is financed by the European Commission under the Sixth Framework Programme. However, this paper expresses the authors' personal views, which are not necessarily those of the DAIDALOS consortium.

0-7695-2520-2/06 $20.00 © 2006 IEEE

Enriching a context-aware system with semantic knowledge representation provides robust and straightforward techniques for describing contextual facts and interrelationships in a precise and traceable manner [5]. Contextual ontologies may provide a thorough representation of knowledge and support reasoning about context information. However, ontologies cannot support capturing, managing, and processing constantly changing information in a scalable manner [6]. On the other hand, traditional context models address many data management aspects using a database-style management of context information. In this framework, a hybrid context modelling scheme is developed in order to integrate the advantages of both approaches and achieve maximum scalability and efficient reasoning and context interpretation in large-scale distributed context-aware systems. This paper is concerned with the creation of a generic context ontology, and a location-based context model. The designed ontology is called COMANTO and describes general context types and interrelationships that are not domain-, application- or situation-specific. The location-based context model proposed focuses on addressing context management challenges in distributed pervasive environments, and is integrated with the COMANTO context knowledge. Our combined modeling approach aims to enable efficient management of context data and allow for a widely applicable context formalism.

The rest of the paper is structured as follows. Section 2 elaborates on the COMANTO context ontology. Section 3 focuses on the formulation of the location-based context model, which is tailored to efficiently support context data management mechanisms. Section 4 presents the rationale behind the hybrid context representation scheme established. Finally, in Section 5 conclusions are drawn, while an outline of the current status is provided.

2. The COMANTO context ontology

Ontologies have been introduced to resolve the knowledge inconsistencies between different domains. They provide a formal and shared conceptualization of a domain that can be communicated between people and remote heterogeneous computing systems. In this framework, ontologies offer quite promising and powerful mechanisms for processing/sharing context and for inferring new knowledge based on available data. Numerous articles in the CA literature have outlined the benefits of using context ontologies and have established various representation schemes in prototype context-aware systems [5][7][8][9][10]. Nevertheless, the development of generic context ontology solutions has been hampered by the overwhelming variety of potential context types that need to be incorporated so that users can enjoy a diverse range of pervasive services. Thus, most of the introduced context ontologies focus on isolated context-aware applications, while, almost no attempt has been made to incorporate and evaluate them in large-scale pervasive systems. The proposed ontology is called COMANTO (COntext MAnagement oNTOlogy), aims towards this direction and is designed to support various stakeholders in sharing, and synchronizing their context knowledge [11]. COMANTO aims to incorporate the advantages of existent approaches in a single solution, and intends to provide a unified, well structured scheme for the semantic representation of context information. In this section the COMANTO ontology is briefly presented, while a service preferences object that illustrates the hybrid context representation potential is provided.

The root class of the COMANTO ontology is the Semantic Context Entity (SCE). Several subclasses that extend the SCE have been identified: Person, PhysicalObject, Legal Entity, Agenda, Activity, Place, Time, Sensor, Network, Service and Preferences. In Figure 1, the defined SCE subclasses are depicted along with their interrelations based on an extension of a preliminary approach presented in [12].

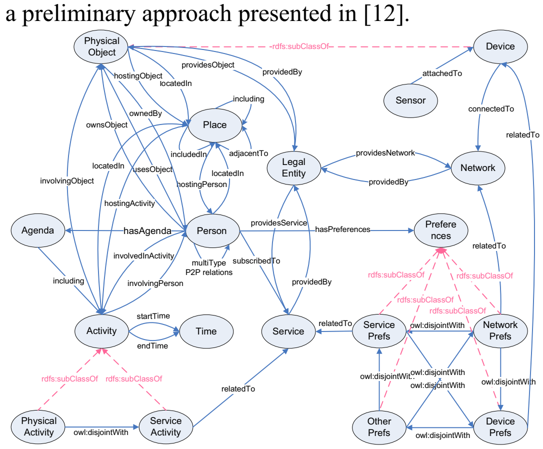

Figure 1. The COMANTO ontology.

Person: The class "Person" is the central entity in the COMANTO ontology and corresponds to all human entities. It offers various datatype properties for incorporating the user related context. For representing the person-to-person associations (e.g. "friendOf"), there is the "multiTypeP2PRelations" object property that is extended by the various P2P properties.

Place: The class "Place" represents the abstraction of a physical spatial place. It offers a set of data properties that traditionally associate a physical location with its symbolic or geographic representation.

Preferences: For semantically representing user preferences in a generic, well-defined and widely accepted manner, four "Preferences" subclasses are distinguished, namely: "DevicePreferences", "ServicePreferences", "NetworkPreferences" and "OtherPreferences". The four preference subclasses are defined to be disjoint with each other.

Activity: The user activity context includes all information relevant to the user's task during the specified task's duration. Two main types are identified: the physical and the service activities. The physical activity (e.g. watching TV) contains all the activities that do not belong to the service activity category. To represent the above concepts, two disjoint "Activity" subclasses have been introduced: the "PhysicalActivity" and the "ServiceActivity".

Agenda: The user agenda context refers to the calendar information of the user. Although, this information is often managed and maintained by a company or a third party service provider, in which case it requires a user subscription, yet we believe that it substantially augments to user's activity modeling. Thus, we introduced the "Agenda" semantic context entity in the COMANTO ontology.

Time: The point in time is crucial information. The time context comprises all information related to the current time and serves as a timestamp for all context information that may change over time.

Physical Object: The notion of "PhysicalObject" is introduced in order to represent artefacts that can not be considered as devices (e.g. furniture). The "Device" class is defined as its subclass.

Sensor: The "Sensor" class is introduced to represent sensors that are used to collect context information. It holds various datatype properties representing sensors' configuration features.

Service: The "Service" class captures information relevant to the services/applications the user has subscribed to.

Network: The "Network" class comprises all information related to the underlying network.

Legal Entity: The "LegalEntity" class represents the corporate actors involved in the pervasive computing supply chain.

There are various object properties that associate the presented semantic context entities of COMANTO, as illustrated in Figure 1, which are described in detail in [11]. The COMANTO ontology has been implemented in OWL, while the Protégé [13] ontology editor and knowledge-base framework has been used.

The established context management infrastructure [14] uses an object oriented context model [3] for communicating the context information among the interested parties, while using the COMANTO ontology to describe the properties and structure of the context model objects. In general, the proposed context model is built upon the notion of an Entity, which corresponds to an object of the physical or conceptual world. Entities may demonstrate various properties, e.g. "location", which are represented by Attributes. Additionally, an Entity may be linked to other Entities via Association objects.

The presented COMANTO ontology is not tied to a specific problem domain. Instead, it attempts to describe generic concepts and their interrelations. In this framework, we have produced an instantiation of the COMANTO for the needs of the DAIDALOS demonstration scenario [11]. These ontology concepts defined the necessary context types and were populated in a semantic context encyclopedia. Such an instance that uses the ontology semantics and the data model to represent a service preferences entity is depicted in Figure 2. It uses the XML language and is dynamically translated to the corresponding context model object.


```xml
<?xml version="1.0" encoding="UTF-8"?>
<ModelObjects>
<Entity>
 <EntityID>cms://context-server:8080/ENTITY/SERVICE_PREFERENCES/28</EntityID>
 <EntityType>SERVICE_PREFERENCES</EntityType>
 <Attributes>
  <Attribute>
   <AttributeID>cms://context-server:8080/ENTITY/SERVICE_PREFERENCES/28/ATTRIBUTE/NAME/1 </AttributeID>
   <AttributeType>NAME</AttributeType>
   <Value>NewsCastPreferences</Value>
   <ActivationStatus>true</ActivationStatus>
  </Attribute>
  <Attribute>
   <AttributeID>cms://context-server:8080/ENTITY/SERVICE_PREFERENCES/28/ATTRIBUTE/PREFERRED_NEWS_TYPE/2</AttributeID>
   <AttributeType>PREFERRED_NEWS_TYPE</AttributeType>
   <Value>any</Value>
   <ActivationStatus>true</ActivationStatus>
  </Attribute>
  <Attribute>
   <AttributeID>cms://context-server:8080/ENTITY/SERVICE_PREFERENCES/28/ATTRIBUTE/QUALITY_LEVEL/3</AttributeID>
   <AttributeType>QUALITY_LEVEL</AttributeType>
   <Value>BestQuality</Value>
   <ActivationStatus>true</ActivationStatus>
  </Attribute>
  <Attribute>
   <AttributeID>cms://context-server:8080/ENTITY/SERVICE_PREFERENCES/28/ATTRIBUTE/SOURCE_URL/4</AttributeID>
   <AttributeType>SOURCE_URL</AttributeType>
   <Value>m4avtest@[ip-v6]:port</Value>
   <ActivationStatus>true</ActivationStatus>
  </Attribute>
 </Attributes>
 <Associations>
  <Association
   identifier="cms://context-server:8080/DIRECTEDASSOCIATION/HAS_PREFERENCES/93" activation="true"/>
  <Association
   identifier="cms://context-server:8080/DIRECTEDASSOCIATION/RELATED_TO_SERVICE/94" activation="true"/>
  <Association
   identifier="cms://context-server:8080/DIRECTEDASSOCIATION/SERVICE_PREFERENCES_OF/74" activation="true"/>
  <Association
   identifier="cms://context-server:8080/DIRECTEDASSOCIATION/HAS_SERVICE_PREFERENCES/46" activation="true"/>
 </Associations>
</Entity>
</ModelObjects>
```


Figure 2. A service preferences context model example object.

The service preference entity illustrated above, represents a subscriber's preferences for a Newscast Service. The respective context entity is identified by a URL-based identifier pointing to the context server that maintains the information under the specific spatial domain. Furthermore, the entity owns a number of attributes that correspond to the preferences' properties (i.e., entity name, preferred news type, quality level and source url). Each attribute carries an activation status, indicating the user current selection. Finally, this preference entity contains some identifiers that point to specific associations objects so as to relate itself with other entities, e.g. with the subscriber's entity through the "hasServicePreferences" association. In our example, the presented associations are tagged as directed and distinguish between the different source and target role. There are also undirected associations that relate peer entities [3]. The presented example demonstrates how the COMANTO concepts are used to create concrete context model objects. The collected data are integrated in the context store using the format imposed by the context model, which inherits the COMANTO context types. Subsequently, an inference engine utilizes the specified ontology instantiation residing in the context semantic encyclopedia [11]. It converts the context objects into COMANTO instances and performs advanced ontology reasoning.

3. A location-based context model

The context repositories must be able to support varying increases in the number of retrieval and update requests and deal with the overwhelming mass of stored context data. The key to building flexible context databases is the distinction between two classes of entities: fixed and mobile. Fixed entities, such as regions, streets, buildings, or even plants, have static geographic coordinates. On the other hand, mobile entities, such as persons or vehicles, are dynamically associated with fixed place entities. These two classes are not only distinguished by their movement profile, but also by the type of databases queries used to retrieve them. While fixed entities are expected to be queried by providing an address or GPS coordinates, mobile ones are typically queried by their unique name, e.g., the name of a buddy: "Where is John?". The heterogeneous requirements on these classes result into two different retrieval mechanisms that are adopted by the implemented context databases.

To optimise the retrieval of fixed context entities, our database implements a location-based retrieval mechanism. Based on the taxonomy of Becker & Dürr [15], a hybrid location model has been implemented that maintains both symbolic entities like streets and buildings, as well as objects identified by plain coordinates only. To speed-up the context retrieval process, entities are arranged in a location-based hierarchy, which is based on the inclusion relation of fixed entities. For example, an entity representing a city is the parent note of all buildings and streets; likewise a building entity is the parent of all room entities. Every DAIDALOS context database maintains a hierarchy of entities, each of them representing an area of the covered domain. The root node of this hierarchy describes the covered area of the domain. Leaf entities describe either places of minimum granularity, or point to other context databases. Figure 3 illustrates an example of this hierarchy. When a client requires an entity that matches a certain coordinate or address, i.e., an entity representing a place that contains the coordinate or address provided, the request is dispatched to the local domain context database. On receiving the request, the database first checks whether the coordinates are within the local domain. If not the database forwards the request to a database server, the domain of which covers a larger geographical area. Thus, in a DNS like fashion, requests are routed through the hierarchy until they reach a server, the domain of which contains the coordinate provided. From this point on, the request is handed down to the sub-domain server that stores the searched entity.

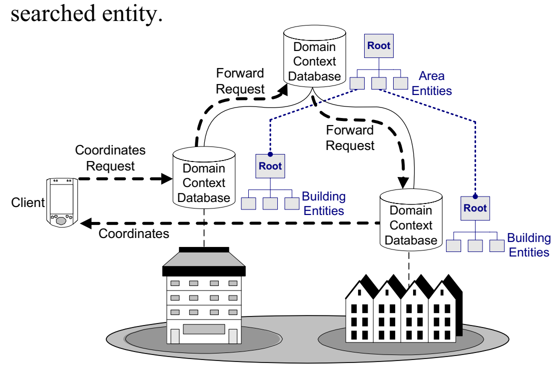

Figure 3. Context database hierarchy.

The proposed approach addresses the scalability requirement by adopting a location centric view of context. Typically, context clients will access data related to the area they are currently located at (i.e. their city or country). The percentage of requests for context information, like for example restaurant menus or outdoor temperature, of fairly remote areas is quite low. Considering this fact, the context management comprises database servers distributed all over the network; each of them being responsible for collecting and maintaining context data that is related to a certain spatial domain, i.e., a preconfigured geographical area or an organization. It has been observed that most context access requests to a database originate from its domain and hence, this approach performs well, utilizing efficiently the resources available.

In contrast to fixed entities, mobile ones are queried explicitly, as in general the requesters know the identity of the entity they are looking for. For example, when a person is looking for his buddies, he will be aware of their identities and will thus be able to provide them to the context database. The implemented approach uses a URL-based context identification scheme [11]. The URL contains the address of a care-of-database that actually stores the entity. As each entity has a care-of-database that stores it's master copy, it is straightforward to retrieve entities if the identifier is known in advance. In a nutshell, following the GSM principles, the care-of-database concept has been used to indicate the location of the master context information instance.

Nevertheless, there is the case where the context requestor is not aware of the URL-based ID, but is only authorised to have access to a "pseudonym" (PNym), e.g. "JohnDoe56". This PNym identifies a context entity, but does not contain any information with regards to its physical location or the database it is stored in. In DAIDALOS, there are secure distributed databases that maintain the context entity PNyms mappings to the actual context URLs [11]. Once a PNym based query is received by the context management system, the PNym is mapped to the URL context identifier and from this point on the information retrieval process is similar to the one described in the previous paragraph.

Mobile entities follow their corresponding physical representatives through the context-database domains. That is, when a user travels to a city, a replica or at least a symbolic link of the user entity's master copy will follow him and will be stored in the city's local context database. Typically, changes to mobile entity objects originate from the domain they are physically geographically located. For instance, a user, who makes changes to his user profile, will by definition be located in the same domain as his representing profile entity. To reduce inter-domain traffic, these changes are first made to the local copy of the profile entity. When these changes are finalized, the local copy is synchronized with the care-of-database. Frequent travelers may even delay updates to the master copy to reduce the expensive inter-domain traffic even further.

4. The rationale behind a hybrid context model

Everything can be considered to be context information. Developers of context management infrastructures are not only challenged by the overwhelming amount of data that needs to be stored in the system's database, they must also standardize a plethora of information types. As opposed to human beings, programs do not exhibit any cognitive abilities that would allow them to interpret information. They are totally reliant on strictly typed data. In this framework, overlapping ontologies and the lack of semantic knowledge of context providers for certain application domains could result in misinterpretations. Nevertheless, a shared ontology model approach offers exceptional benefits, similar to the ones brought out by the successful use of standards within various computer science domains, which operate as a kind of shared ontology or model.

Context ontologies address the need of applications to access a thorough, widely accepted and formal representation of knowledge in order to interpret and reason about context information and act accordingly. However, they are not designed for capturing and processing constantly changing information in dynamic environments in a scalable manner, but for statically representing the knowledge of a domain [6]. On the other hand, traditional relational context models provide a database-style management of context data and offer standard interfaces for applications to query context information or receive notifications on context changes. The latter scheme addresses several critical issues in the design of efficient context management systems, such as the provision of secure context storage, real-time dissemination and update efficiency, caching and replication of context-information across multiple administrative domains.

The gradual increase of the amount of context data, the multitude of context providers and sources involved, the requirement for real-time context retrieval and dissemination anytime and anyplace, the need for seamless mobility across devices, and the heterogeneity of devices used, lead to the conclusion that a dual management approach should be adopted by context systems that distinguishes between context object and context knowledge management. Our hybrid context modeling approach is built on this principle. On the one hand, a location-based context model has been designed for distributing and managing context data in the pervasive service platform. On the other hand, the COMANTO ontology is utilized to describe contextual types and associations between entities, thus capturing a domains' context knowledge, supporting efficient context reasoning both over local and distributed information, whilst serving as a public context semantic encyclopedia. In this framework, based on the COMANTO representation scheme, context sources monitor the environment and update the instances of the established location-based context model.

A combined approach of spatial context models and contextual ontologies has also been proposed in [6]. This preliminary work examines the advantages and disadvantages of context models and ontologies, suggesting that context management is handled by plain models, while context ontologies cope with information representation and reasoning. Our hybrid context modelling approach goes beyond this work, since it introduces the COMANTO ontology combined with an innovative location-based context model [3]. The integration in our hybrid approach is vertical and evident in all service provision layers. Furthermore, the distinction between fixed and mobile entities optimises the performance of the query handling mechanism with regards to the precision and recall parameters [16], enhances the scalability of the context repositories with regards to the required storage capabilities [11] and reduces the lifecycle of the context retrieval and update processes [16].

5. Conclusions

CA creates new business opportunities for new attractive human-centric applications and facilitates greatly the human-computer interaction. It aims to optimise the service experience of users and relieve them from providing information that already exists in other parts of the global system or can be sensed or inferred. Nevertheless, CA can not be achieved before an end-to-end uniform approach for modeling context information has been established, across a multitude of administration domains.

This paper focuses on the design of a context modeling approach that aims to combine the semantic advantages of context ontologies and the maintenance, distribution and administrative facilities of a location-based context model. The benefit of this approach is twofold. On the one hand, the context information is managed, disseminated and federated in a well-defined, reusable and scalable manner exploiting the location-based context model built, while on the other hand a flexible, interoperable, and generic context knowledge representation scheme is established by the COMANTO ontology. In December 2005, the DAIDALOS pervasive service platform that implements the hybrid approach proposed has been demonstrated and evaluated against various criteria. Its validation will hopefully contribute to the introduction of pervasive services in the wide market.

6. References

[1] M. Satyanarayanan, "Pervasive computing: vision and challenges", *IEEE Pervasive Computing*, 8(4), Aug. 2001, pp. 10-17.

[2] S. Xynogalas, et al., "Context Management for the Provision of Adaptive Services to Roaming Users", *IEEE Wireless Communications*, 11(2), Apr. 2004, pp. 40-47.

[3] M. Strimpakou, et al., "Context Modelling and Management in Ambient-aware Pervasive Environments", *Int. Workshop on Location and Context-Awareness* (LoCA 2005), Oberpfaffenhofen, Germany, May 2005.

[4] K. Henricksen, and J. Indulska, "Developing context-aware pervasive computing applications: Models and approach", *Pervasive and Mobile Computing Journal* (in Press), 2006.

[5] T. Strang, and C. Linnhoff-Popien, "A Context Modeling Survey", *Workshop on Advanced Context Modelling, Reasoning and Management under the 6th Int. Conf. on Ubiquitous Computing* (UbiComp 2004), Nottingham, England, Sept. 2004.

[6] C. Becker, and D. Nicklas "Where do spatial context-models end and where do ontologies start? A proposal of a combined approach", *Workshop on Advanced Context Modelling, Reasoning and Management under the 6th Int. Conf. on Ubiquitous Computing* (UbiComp 2004), Nottingham, UK, Sept. 2004.

[7] H. Chen, T. Finin, A. Joshi, and L. Kagal, "Intelligent Agents Meet the Semantic Web in Smart Spaces", *IEEE Internet Computing*, 8(6), Nov.-Dec. 2004, pp. 69-79.

[8] X. H. Wang, D. Q. Zhang, T. Gu, and H. K. Pung, "Ontology Based Context Modeling and Reasoning using OWL", *2nd IEEE Int. Conf. on Pervasive Computing and Communications 2004* (PerCom 2004), Workshop on Context Modeling and Reasoning, Orlando, USA, Mar.2004.

[9] P. Korpipaa, J. Mantyjarvi, J. Kela, H. Keranen, and E. Malm, "Managing Context Information in Mobile Devices", *IEEE Pervasive Computing*, 2(3), Jul.-Sept. 2003, pp. 42-51.

[10] M. Khedr, and A. Karmouch "ACAI: Agent-Based Context-aware Infrastructure for Spontaneous Applications", *Journal of Network and Computer Applications*, 28(1), Jan. 2005, pp. 19-44.

[11] I. Roussaki (edt.), "Daidalos Deliverable D431, Context awareness in Daidalos: Design and implementation", *DAIDALOS Integrated Project* (FP6-2002-IST-1:506997), Sept. 2005.

[12] S. Xynogalas, I. Roussaki, M. Chantzara, and M. Anagnostou, "Context Management in Virtual Home Environment Systems", *Journal of Circuits, Systems and Computers*, 13(2), Apr. 2004, pp. 293-311.

[13] J.H. Gennari, et al., "The evolution of Protégé: an environment for knowledge-based systems development", *Int. Journal of Human-Computer Studies*, 58(1), Jan. 2003, pp. 89-123.

[14] H. Williams, et al., "Context Awareness and Personalization in the Daidalos Pervasive Environment", *IEEE Int. Conf. on Pervasive Services* (ICPS 2005), Santorini, Greece, July 2005.

[15] C. Becker, and F. Dürr, "On location models for ubiquitous computing", *Personal and Ubiquitous Computing*, 9(1), Jan. 2005, pp.20-31.

[16] C. Pils, I. Roussaki, and M. Strimpakou, "Location-based Context Retrieval and Filtering", Submitted to the *Int. Workshop on Location and Context-Awareness* (LoCA 2006), Dublin, Ireland, May 2006.

Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06)
0-7695-2520-2/06 $20.00 © 2006 IEEE