Towards an Ontology and DHT-based publish/subscribe scalable system

Amina Chaabane [1,2,3,4], Wassef Louati [1,2], Mohamed Jmaiel [1,2], Jorge Gómez-Montalvo [3,4,5], Codé Diop [3,4], Ernesto Exposito [3,4]

[1] University of Sfax, ENIS, ReDCAD Laboratory, Route de la Soukra, B.P. 1173, 3038

[2] University of Sfax, Sfax, Tunisia

[3] CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France

[4] Univ de Toulouse, INSA, LAAS, F-31400 Toulouse, France

[5] Facultad de Matemáticas, UADY, México

{amina.chaabane, jgomez, cdiop, eexposit}@laas.fr ; louatiw@gmail.com,Mohamed.Jmaiel@enis.rnu.tn

Abstract—In the context of Publish/Subscribe systems (Pub/Sub), Peer-to-Peer (P2P) solutions based on Distributed Hash Tables (DHT) offer efficient functionalities such as event routing flexibility, scalability, load balancing and fault tolerance. For event routing, it uses event identifiers computed by event encryption. Likewise, Pub/Sub systems based on ontologies offers well adapted semantic expression capabilities allowing consumers and providers to easily describe services' requests and properties. Thanks to the ontologies, semantic reasoning allows to provide enhanced event discovering. Indeed, DHT-based systems are based on traditional matching algorithms while in ontology-based systems, users can use various literals semantically equivalent but syntactically different expressions in order to describe their requirements. However, both approaches present limitations in terms of: semantic expression restrictions for the first one and scalability issues for the second one. Our solution consists in combining the advantage of both of them, by taking advantage of the semantic capabilities of ontologies at local domain level and the scalable P2P-DHT solutions at inter-domains level. This paper proposes an approach based on a community-wide ontology in order to allow publishers and subscribers to use a common semantic space to characterize production and consumption of resources and services. Based on this ontology, a P2P network based on distributed reasoners (i.e. inference engines) will be able to evaluate if a specific content belongs to an ontology-based community topic. Thus, we use these topics to compute keys for routing events in a DHT.

Keywords: Pub/Sub systems, P2P networks, DHT, Ontologies, reasoners.

I. Introduction

Event-based systems require an efficient communication model that usually is deployed on large distributed infrastructures. These systems ensure interaction between two actors: consumers and providers (based on consumers' needs) [5]. These actors can be represented by a simple user or a personal network's (PN) user which is located in his cluster composed of personnel devices [13]. Consumers are concerned on expressing their interests in the form of a subscription (i.e. filter). Consumers need to personalize their subscription according their profiles and preferences. A provider publishes some data to the intermediate event service which is responsible to sort and filter publication only to interested consumers. Pub/Sub systems are classified according to event service topology which describes the brokers' relationships. This topology is classified on three basis categories: the hierarchical, the P2P cyclic (structured, unstructured) and the P2P acyclic [4]. The preferred topology is P2P cyclic as much as it ensures scalability. In particular, the structured P2P cyclic takes the opportunity of using DHT deployed on large infrastructure to facilitate event routing. Based on DHT, we must compute an event ID as a key to store and to lookup each event in and from the root node (responsible node) respectively. The first class is the topic-based system in which subscriptions contain only the name of a class of messages. Each class describes a topic which regroups objects expressing an affiliation through their subject. The second class of systems is the type-based subscription scheme. It is very close to the subject-based scheme. Instead of using additional subjects to logically group data, the type-based scheme uses the objects types to differentiate between logical groups. These two classes of Pub/Sub systems offer only restricted expressiveness. Finally, the content-based Pub/Sub system fills this gap by introducing a subscription scheme based on the content of the published information. With this class, Pub/Sub systems are most expressive and allow users to obtain just events to which they are interested on. Subscribers can define, by using a triple (attribute, operator, value), the criteria that the content of a published event should satisfy. This is the basic format subscription of the content-based system. Other formats are proposed with XML, RDF and OWL in order to enhance the event semantic. Nevertheless, content-based Pub/Sub are not simple to be implemented with DHT since it is necessary to define standardized event IDs. Thus, there is not enough semantic in current content-based Pub/Sub system based on DHT.

In this paper, we handle the expressiveness limits of Pub/Sub systems such as Scribe built on top of Pastry nodes [6, 16]. Scribe is a well scalable topic-based system. However, it does not support semantic subscriptions and publications. Despite of the fact that some solutions handle expressiveness limits in Scribe, these solutions are not sufficient when dividing the content into several topics for its submission to a topic-based system. To support more expressiveness, we propose an ontology-based approach with a numerically encoding procedure applied to the used ontology allowing ontology-based event to be routed through DHT nodes and a semantic matching for subscription discovery. This procedure is inspired from EASY approach proposed by S. Ben Mokhtar et al. [14].

The remainder of this paper is organized as follows. Section II presents a broad overview of related works. In section III, our solution aimed at coping with expressiveness limits of Pub/Sub systems is presented. Section IV presents several use cases and how our method ensures semantic matching efficiency in DHT-based Pub/Sub system. The last section concludes this paper and gives future directions of this work.

II. Related Work

There has been little research on semantic Pub/Sub system based on DHT in scientific literature. Several research works have introduced different solutions to improve scalability and semantic matching. We find two types of approaches which generally cope these two Pub/Sub properties. The first approach satisfies the scalability property using structured P2P event service on DHT but fails in maintaining the semantic. The second kind of approach improves the semantic matching based on ontologies but it loses the scalability of the system.

A number of research works have been conducted in the area of DHT-based Pub/Sub system such as Scribe [6], p2p-ToPSS [18], Willow [19] and PastryStrings [1] and Meghdoot [11]. Scribe is built on Pastry DHT to construct a multicast tree for each topic. It is scalable and repair peer failure but the main point of failure is the expressiveness. In fact, it is topic-based with subscription in the simple form: “topic=x”. P2P-ToPSS extends Scribe by enhancing subscription form only for numeric values. It defines schema to fix numeric value on interval. Then, when user defines subscription with numeric range, it uses defined schema for dividing this range into possible values which will be simple topics. This approach presents some drawbacks. First, P2P-ToPSS overloads Pastry and several multicast trees when dividing one subscription on several topics. Second, it is efficient neither for open interval nor for real values. Thus, it is not expressive and scalable. PastryStrings handles String and numerical subscriptions with simple comparison operators. Only, with limited number of messages, this system is scalable but it is still semantically inefficient. Meghdoot is a content-based system over structured P2P networks. It is built on CAN DHT [15]. Meghdoot subscription is defined through attributes of a defined schema S= {A_1, A_2, . . . , A_n}, where each attribute A_i has a name, type and domain, and can be described by the set {Name: Type, Min, Max}. Thus, semantic subscription is not allowed in Meghdoot and all content-based Pub/Sub over structured P2P networks. To overcome expressiveness problem, A. Carzaniga et al. [4] propose SIENA as a content-based system with hierarchical and unstructured P2P topology. SIENA improves expressiveness with various subscription types as integer, float, string, byte, etc. and various comparison operators. The main shortcoming in SIENA is scalability and load balancing between event servers. Then, J. Keeney et al [12] extend SIENA with ontology-based subscription and publication and introduce simple covering relationships with some operators to reduce the subscription matching and routing overhead. Thus, scalability and load balancing problems persist. J. Skovronski et al. [17] use ontology to describe published data and SPARQL query as subscription. They classify exchanged data into domains as topics. For each domain, they define an ontology model and a responsible agent. This approach proposes a centralized router which maintains all agents responsible to various domains. The major advantage of this work is the semantic description using ontology model. However, this approach is centralized which inhibits the scalability.

In this paper, we propose a scalable solution based on semantic encoding algorithm to overcome limits raised by the two classes of existing solutions. Our approach satisfies semantic and scalability properties within an efficient content-based Pub/Sub middleware.

A. Background on Scribe and Expressiveness Limits

The Scribe multicast tree is built over Pastry DHT. Each broker is presented by a Pastry peer which is identified by an unique digit sequence (128 bits).

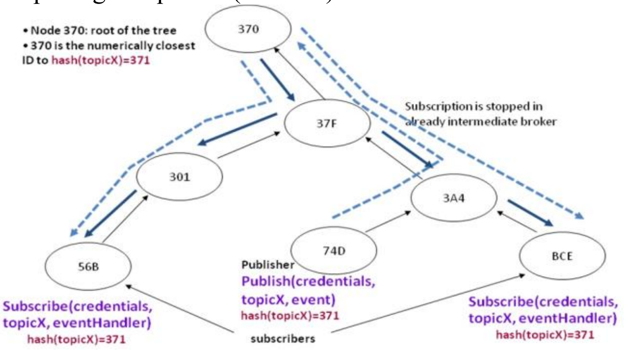

Figure 1. Scribe multicast tree

In subscription phase, users create different topics to which other users can subscribe (see Figure 1). When creating a new topic, a new multicast tree will be created and a new broker will be selected as responsible for the new topic. The selected broker must have the closest ID to the subscription’s key. While going from the subscriber to the responsible broker, a new branch of the tree is added. Each branch is made up of intermediate brokers. When a topic is already created, if a subscription reach a broker already reached, then the subscription doesn’t continue the path worm the target (responsible broker). In publication phase, the created multicast tree is used as follows: the published topic reach the responsible broker (root) which will be responsible to forward the notification to interested users through branches of its appropriate tree.

The expressiveness shortcoming in Scribe is very apparent. In fact, it only allows filter with equality operator and an exact matching in view of the fact that DHT encodes event for routing in P2P. Besides, the Scribe routing does not consider the semantic equality e.g. if a user subscribes to “topic=video” and later a publication with “topic=film” is produced, the subscriber can not be notified in spite of the equivalence between “film” and “video”.

III. Ontology-Based P2P Pub/Sub System

In this section, we detail our approach based on a multimedia delivery use case scenario. Our approach follows three steps. First, we define the largest taxonomy possible with ontology description by extending an existent domain ontology as the Multimedia Web Ontology Language that we use in this example. This ontology description must be well structured for organizing services according to their semantic similarity. This allows efficiently concept and class encoding for event routing and localization through DHT. Second step consists in instantiating this ontology for subscription or publication events. Third, based on the domain ontology, the topic classes will be defined by axioms expressing the properties that need to be satisfied by the events belonging to the topics. Inference engines use these descriptions in order to dynamically compute topic publishing and subscription. Once the topic is determined, we follow the routing phase on DHT. We detail each step in following sections.

A. Domain Ontology Definition

We define a domain ontology that covers the application domain including the definitions of several concepts organized in a large taxonomy (see Figure 2). In the domain ontology, classes are structured according to inheritance relationships similarly to the object-oriented inheritance hierarchies. In a multimedia scenario, the root node of ontology graph represents the multimedia element which is more generic literal. Then, we find multimedia element types which are generic and can match a larger number of multimedia content compared with the types contained in the rest of the ontology graph. On the contrary, leaf nodes of a graph contain multimedia types that are said to be more specific than the other types. Thus, when a user subscribes to the topic “video”, then a publisher produces a “film”, our system must detect that the “film” is a “video” and thereafter notify the subscriber. In this way, DHT nodes that are responsible for the specific topic (child topic) must know and communicate with others nodes that are responsible for parent topics. Consequently, the proposed structure is proven. Besides, this ontology structure is feasible (possible in real life domain) especially if it is defined from a well taxonomy. When domain ontology is well structured and defined, it will be shared with all users. Each user (subscriber or producer) must create an individual as a subscription or a publication event.

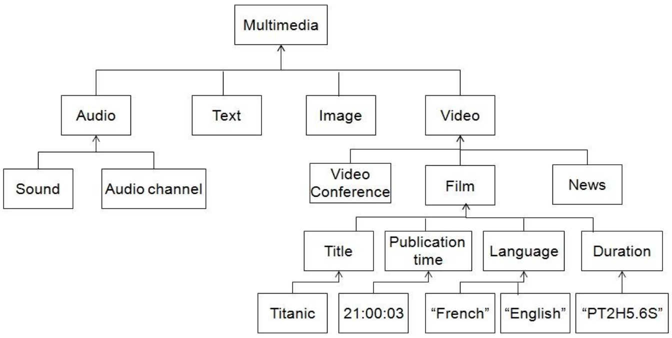

Figure 2. Ontology instance example

Then, we use a reasoner to infer topics from described composed content of publication/subscription for each user (locally). Thus, our approach combines semantics and scalability, which are provided by the ontology and DHT respectively. The following step consists on routing subscription/publication according to topic ID. In the following section, we detail topic ID calculation.

B. Semantic Encoding Using Prime Numbers

For the DHT routing, we have said that DHT nodes that are responsible for a specific topic (child topic) must know and communicate with others nodes that are responsible for generic topics. For this reason, ontology semantic encoding is crucial for DHT nodes communication. In this step, we have followed the approach proposed by S. Ben Mokhtar et al. [14] for ontology encoding. As shown in Figure 3, we use prime numbers to assign IDs to topics in the domain ontology. We start enumeration from the parent to the leaf with prime numbers. We grant prime number to parent as an ID. And for children, we grant the multiplication of parent IDs with the next prime number. Thus, the ID_i of class C_i is given by the expression: $ID_i = \pi_i$ x p_i when π_i is the multiplication of parent class IDs of class C_i (underlined numbers in Figure 3) and p_i is the prime number of C_i. Which enables us to deduce from a class ID its parents (generic topic) by simple computing divisors of the ID which are prime numbers. Once the class ID is calculated, it will be used for routing on Scribe.

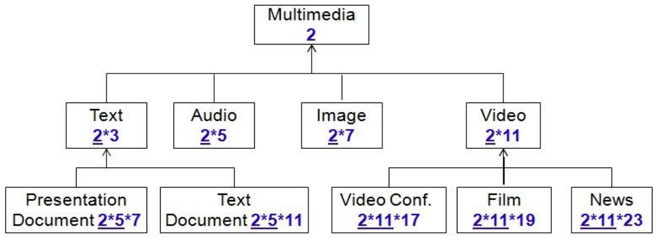

Figure 3. Semantic encoding ontology

C. Routing phase on Scribe

For the subscription/publication phase, consumers use a GUI for ontology description. Then, we use a reasoner to find topics. After that, we search the corresponding class ID. Thus, this class ID will be used as a key and the ontology description as a value and we use Scribe for subscription and publication routing as explained in background section. We improve Scribe by modifying publication phase. In fact, when publication reaches responsible node (root) in the multicast tree, the root can handles his ID:

- If the ID is a prime number then it is a leaf class in the domain ontology and we finish Scribe notification phase normally.
- If the ID is not a prime number then it computes its divisors which are prime numbers. From these prime number divisors, we can find IDs of more specific classes. These prime number divisors are IDs of more specific classes. Therefore, the root sends the publication to roots responsible to the specific classes. In fact, these roots can have subscribers to more specific topics and this publication matches it.

IV. USE CASES

In this section several contexts where our solution could be used will be introduced and illustrated.

D. Home Networks

The explosion of mobile devices with communication service capabilities has contributed to the apparition of ubiquitous multimedia scenarios where users can easily create ad hoc multimedia sessions. Moreover, the increasing use of social networks has facilitated the apparition of new scenarios including new ways of sharing and communicating multimedia information. Within the vast diversity of ubiquitous multimedia scenarios, we have found that multimedia scenarios in home networks are particularly important because users produce, consume, and share a considerable amount of multimedia information in home networks. The UPnP Forum [20] and DLNA [2] have envisioned such scenarios and they have proposed a set of technologies such as [8, 7, 9] in order to facilitate the integration of home devices in the digital living experience. Users can also produce, consume, and share multimedia information when they are away from their home networks. Furthermore, users expect as good quality of service from their multimedia services as when they are in their home networks. The Extended Home concept deals with scenarios where home based services are accessible to the home user whether he is inside or away from his own home. In an extended home, the user not only has access to his services from his car, office, or other homes but also he has similar experience no matter his location. Projects like Feel@Home [3] and EnComPas [21] have targeted and developed such kind of scenarios. Recently, the UPnP Forum has proposed the UPnP Remote Access Architecture [10] which aims to allow generic UPnP devices, services and control points to remotely interact with the corresponding UPnP devices, services and control points in a UPnP home network.

We have identified two scenarios in which semantic description (based on ontologies) facilitates searching of multimedia content according to user preferences. First scenario (Fig. 4) shows the home user uses a graphic user interface to provide a description about the topic in which he is interested. The GUI shows topic's options and parameters according to the content of the ontology-based models (1). Once the user has provided its preferences, this information is used by a reasoner and rule engine in order to infer a list of topics which satisfy user's demand (2). The results obtained by the reasoner and rule engines are presented to the user through a GUI from which he can select the desired content (3).

In the second scenario illustrated by Figure 5, the user makes a wide search which includes multimedia content located not only at his home but in other home networks. In this case, the proposed P2P DHT solution helps to find a route to the home networks containing the desired topic(s). In Figure 5, we can see that the user is able to obtain desired topics stored in other homes by using the UPnP DHT (4). In the Home Network B, the local reasoner and rule engines can refine the user search (5) based on the semantic topic description provided by the user in (1).

E. Service Oriented Architectures

Other potentials use case scenarios are the Service Oriented Architecture (SOA). SOA is an architectural framework or referential model for building software systems based on distributed services which may be offered by different service providers [22]. In a SOA framework, a service exposes its functionalities in the form of services that can be reused across different applications, and services consumers are loosely coupled to the service producers and can bind to the service at development-time or runtime. To implement this binding, several methods exist. But for each case, a service repository which provides the required facilities to discover and use services is needed [23]. The service repository stores the details about the services that can be invoked and how to invoke them. We present hereafter two SOA scenarios in order to illustrate the use of our proposed solutions.

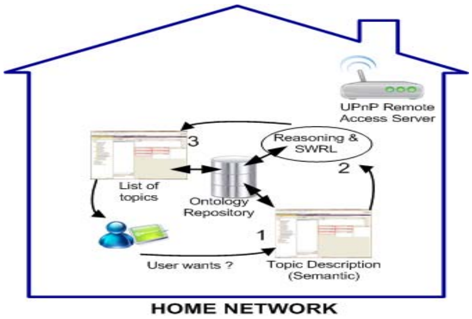

Figure 4. Topic access by providing semantic description

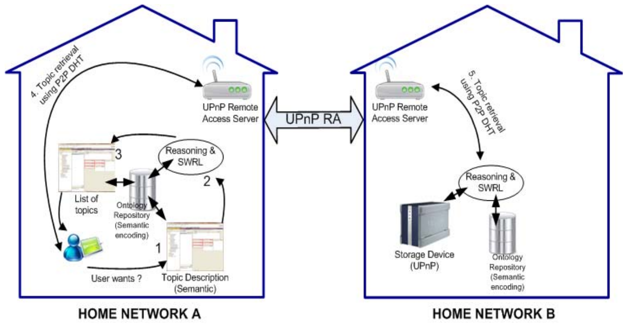

Figure 5. Remote access of topic using P2P DHTUnits

1) UDDI

Universal Description Discovery & Integration (UDDI) is the definition of a set of services supporting the description and discovery of (1) business, organizations, and other Web services providers, (2) the Web services they make available, and (3) the technical interfaces which may be used to access those services [24]. By using our proposed approach (see Figure 6), a multimedia service provider can use the defined ontology to register to the UDDI repository and to publish the semantic expression of capabilities and properties of the provided services. Thank to DHT, many service providers may register to the UDDI repository. By using semantic encoding prime number, scalability issues will be taken into account. When customers need a service, they will invoke the UDDI repository by describing service requests and properties. The resulting discovery service will be more efficient by taking into account service semantic while using DHT for reducing response time.

2) JNDI

Another way to bind service user and service provider is by replacing synchronous messages by asynchronous message. It aims at dealing with synchronization and reliability. In the

Publish/Subscribe model, entities check on a certain topic. JNDI (Java Naming and Directory Interface) repository is used to find out the requested topic. JNDI is a Java API that offers a naming service and a directory interface. The discovery of topics will be made from the JNDI. So, it is possible to use our proposed ontology as a naming service to associate previously a clear semantic to the topics in the case of federated domains. By using this ontology combined to the DHT proposition topic search, message routing efficiency will be improved.

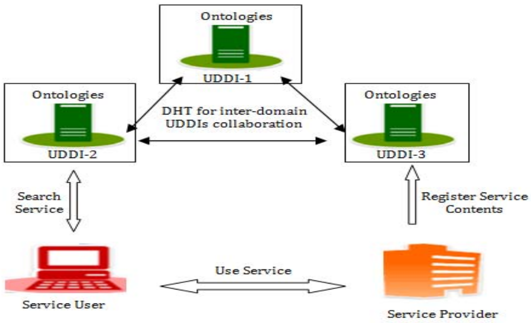

Figure 6. UDDI with ontologies and DHT

CONCLUSIONS AND PERSPECTIVES

In this positioning paper we have proposed a solution aimed at offering semantic richness and scalability guarantees for large publish/subscribe systems. Our solution consists in taking advantage of semantic capabilities of ontologies at local domain level and scalable P2P-DHT solutions at inter-domains level. Our approach is based on using well-established and adopted community-wide ontologies in order to allow publishers and subscribers to use a common semantic space to characterize production and consumption of resources and services. Based on this ontology, a P2P network based on distributed inference engines are able to evaluate if specific content belongs to an ontology-based community topic. These topics are used to compute keys for routing events in a P2P-DHT system.

In order to illustrate the advantages and the feasibility of our approach, we have developed two kind of generic use case scenarios: home networks and service oriented architectures. In home network scenarios, consumers and providers uses a common multimedia web ontology in order to describe intra-home multimedia content publishings and subscriptions. In such way, a P2P-DHT system will allow to share multimedia content between large inter-home networks. The second scenario illustrates how UDDI or JNDI distributed directories could be defined using domain-level ontologies and inter-domain P2P-DHT systems in the framework of large SOA architectures. Our work is still in progress and the current and the future efforts will be intended to completely develop and deploy the introduced use case scenarios in the framework of 3 PhD thesis during the FP7-ICT IMAGINE european project.

ACKNOWLEDGMENT

This work has been supported by FP7-ICT IMAGINE research and development project, funded by the European Commission under the “Virtual Factories and Enterprises” (FoF-ICT-2011.7.3, Grand Agreement No: 285132).

REFERENCES

[1] I. Aekaterinidis and P. Triantafillou. Pastrystrings, “A comprehensive content-based publish/subscribe dht network,” In ICDCS, page 23. IEEE Computer Society, 2006.

[2] D. L. N. Alliance, “DLNA Overview and Vision Whitepaper,” White paper, DLNA, 2007. Available online (23 pages) http://www.dlna.org/news/DLNA white paper.pdf.

[3] M. Bel-Martin, G. Maestro-Molina, M. Mahdi, and O. Dugeon, “ Digital Home Networking, chapter The Feel@Home System,” ISTE Ltd, Wiley, 2011.

[4] A. Carzaniga, D. S. Rosenblum, and A. L. Wolf, “ Interfaces and algorithms for a wide-area event notification service,” 2000.

[5] A. Carzaniga, D. S. Rosenblum, and A. L.Wolf, “Design and evaluation of a wide-area event notification service,” ACM Trans. Comput. Syst., 19:332–383, August 2001.

[6] M. Castro, P. Druschel, A.-M. Kermarrec, and A. Rowstron, “Scribe: A large-scale and decentralized application-level multicast infrastructure,” IEEE Journal on Selected Areas in Communications (JSAC), 20, 2002.

[7] U. Forum, “UPnP AV Architecture for Universal Plug and Play Version 1.0,” Sept. 2008.

[8] U. Forum, “UPnP-Device Architeture 1.1,” Oct. 2008.

[9] U. Forum, “UPnP-QoS Architecture:3 For UPnP Version 1.0 Status: Standardized DCP,” Nov. 2008.

[10] U. Forum, “Remote Access Architecture 2,” Apr. 2011.

[11] A. Gupta, O. D. Sahin, D. Agrawal, and A. E. Abbadi, “Meghdoot: Content-based publish/subscribe over p2p networks,” in H.-A. Jacobsen, editor, Middleware, volume 3231 of Lecture Notes in Computer Science, pages 254–273. Springer, 2004.

[12] J. Keeney, D. Roblek, D. Jones, D. Lewis, and D. O’Sullivan, “Extending siena to support more expressive and flexible subscriptions,” in R. Baldoni, editor, DEBS, volume 332 of ACM International Conference Proceeding Series, pages 35–46. ACM, 2008.

[13] W. Louati and D. Zeghlache, “Personal overlay networks management using a p2p-based publish/subscribe naming system,” in Proceedings of the 16th Euromicro Conference on Parallel, Distributed and Network-Based Processing (PDP 2008), pages 95–99,Washington, DC, USA, 2008. IEEE Computer Society.

[14] S. B. Mokhtar, D. Preuveneers, N. Georgantas, V. Issarny, and Y. Berbers, “Easy: Efficient semantic service discovery in pervasive computing environments with qos and context support,” Journal of Systems and Software, 81(5):785–808, 2008.

[15] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, “A scalable content-addressable network,” SIGCOMM Comput. Commun. Rev., 31:161–172, August 2001.

[16] A. I. T. Rowstron and P. Druschel, “Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems,” in Middleware’01, pages 329–350, 2001.

[17] J. Skovronski and K. Chiu, “Ontology based publish subscribe framework,” in iiWAS, pages 49–58, 2006.

[18] D. Tam, R. Azimi, and H. arno Jacobsen, “ Building contentbased publish/subscribe systems with distributed hash tables,” in International Workshop On Databases, Information Systems and Peer-to-Peer Computing, pages 138–152, 2003.

[19] R. van Renesse and A. Bozdog, “Willow: DHT, aggregation, and publish/subscribe in one protocol”.

[20] The UPnP Forum. http://www.upnp.org/

[21] The EnComPAs (Enabling Community Communications - Platforms and Applications) project website, http://encompas.org/

[22] Reference Model for Service Oriented Architecture1.0, OASIS Standard, October, 2006.

[23] D. Krafzig, K. Banke, D. Slama, “Enterprise SOA: Service-Oriented Architecture Best Practices,” Prentice Hall, November 09, 2004.

[24] “UDDI Spec Technical Committee Draft 3.0.2,” OASIS Standard, October 2004.