Emergent Semantics and Cooperation in Multi-Knowledge Environments: the ESTEEM Architecture*

The ESTEEM Team
http://www.dis.uniroma1.it/∼esteem/
esteem@dis.uniroma1.it

ABSTRACT

In the present global society, information has to be exchangeable in open and dynamic environments, where interacting peers do not necessarily share a common understanding of the world at hand, and do not have a complete picture of the context where the interaction occurs. In this paper, we present the ESTEEM approach and the related peer architecture for emergent semantics in dynamic and multi-knowledge environments. In ESTEEM, semantic communities are built around declared interests in the form of *manifesto ontologies*, and their autonomous nature is preserved by allowing a shared semantics to naturally *emerge* from peer interactions.

1. INTRODUCTION

In the present global society, all major organizations have decentralized structures and their information systems handle a variety of information sources. Actually, the problem of how to provide transparent access to heterogeneous information sources while maintaining their autonomy already appeared decades ago, and has been almost solved by information integration techniques, where interaction between clients and data sources happens through a centralized access point and uniform query interfaces give users the illusion of querying a homogeneous system [25, 29]. However, these techniques work under certain hypotheses, including moderately static scenarios, shared understanding of the domain of interest (in form of global schema or ontology), a closed, or at least access-controlled, set of participating sources. All these hypotheses do not hold anymore in the current evolving web of millions of autonomous information nodes (peers) which need to cooperate by sharing their resources (such as data or services). Information has thus to be exchangeable in open and dynamic environments, where interacting peers do not necessarily share a common understanding of the world at hand, and do not have a complete picture of the context where the interaction occurs. Conversely, they dynamically build new information or knowledge, create new semantic communities and establish a new form of context-aware semantic interoperability, based on dynamic trustful agreements on common interpretations within the scenario of a given task, that we refer to as "emergent semantics" [23]. At present, only few research efforts have been produced to face the new requirements of emergent semantics. Difficulties mainly arise due to the highly dynamic nature of peers' interoperability, the lack of any agreed-upon global ontology, as well as the need of distributing the computation to the single nodes when processing queries and composing services in a P2P environment. Hence, new solutions are needed for such issues as agreement or consensus construction, trust and quality management, P2P infrastructure definition, query processing and dynamic service discovery in a context-aware scenario.

Example scenario. According to the previous considerations, a significant example is constituted by medical data that are stored in a large amount of disparate sources, whose quality and trustworthiness is often not directly available. Existing systems allow querying of locally stored information but doctors also need to access data stored in a vast number of heterogeneous distributed sources. This is especially true in the case of diseases which are difficult to diagnose or for which on-going research offers continuous advances in treatment procedures and available pharmacological support. Obtaining this kind of information currently requires time-consuming searches and one-by-one querying of databases available over the web. Let us consider the following scenario. A doctor working in a small hospital in Central Africa, has a patient with a complex clinical condition. In fact, he suffers from malaria but he also has a strong adrenal insufficiency and he is weakened by a chronic disease due to inadequate nutrition. Such a clinical condition could cause side effects to the standard malaria cure.

First (step 1), the doctor performs a web search on a site that she knows dealing with the particular pathology she is looking for. The web search provides lot of results, most of which poorly focused on the problem. After a laborious screening task to detect a helpful result, the doctor has the problem of understanding how trustable it is. Second (step 2), the doctor performs a new web search in order to check if other sites provide the same directions. It may happen that different sites provide conflicting results. Finally (step 3),

*This paper has been partially funded by the ESTEEM PRIN project of the Italian Ministry of Education, University, and Research.

International Workshop on Semantic Data and Service Integration (SDSI'07), September 23, 2007, Vienna, Austria.

the doctor has also to check that the provided results are up-to-date. For instance, if a drug is 'unknown' in the queried sites, she has to verify the availability of recent publications on that drug, e.g. by querying PUBMED.

The described scenario has many weaknesses:

1. The doctor must perform a manual search for all the sources that may provide the interesting solution, thus wasting time and having to rely on a personal knowledge about the sites to be queried.
2. The doctor has no way to verify the trustworthiness of a result.
3. In Step 2, the doctor has no way to evaluate the quality of the results provided by different sites.
4. In Step 3, a new web search must be performed, which, again, may be very time consuming.

Means to effectively cut cost and time in diagnosis and decision-making would require the possibility for a doctor to input a patient condition and obtain from the system all relevant information about it, from the genes that may contribute to causing it, to the symptoms and possible treatments. This requires to *integrate* disparate sources that are often not known in advance and it requires the availability of suitable web services apt to explore sources and obtain useful information in a flexible way. Furthermore, some specific scenarios require that integrated information is accessed through non-conventional devices. For example, it is essential that, when medical personnel is dispatched in remote areas in response to various emergencies like the spreading of epidemic diseases, they are able to remotely access data and to exploit the expertise of specialists in different fields. In these scenarios, doctors need to access information that may help them in formulating correct diagnoses and choosing adequate treatments through easily carried devices like PDAs or last generation mobile phones. As a matter of fact, different contexts lead to different needs and interests, thus context-aware resource selection becomes essential in this scenario.

In this paper, we present the ESTEEM approach and the related peer architecture for emergent semantics in dynamic and multi-knowledge environments like the one described above. In ESTEEM, semantic communities are built around declared interests in form of manifesto ontologies, and their autonomous nature is preserved by allowing a shared semantics to naturally *emerge* from peer interactions. Specific key contributions of ESTEEM regard: i) the definition of a comprehensive peer architecture for enabling effective ontology-based data/service discovery under dynamic and context-dependent requirements, ii) the use of a shuffling-based P2P infrastructure to support peer interactions and the self-formation of peer semantic communities, iii) the use of an ontology-based matchmaker for semantic affinity evaluation of the knowledge provided by different peers, and iv) the specification of trust-aware P2P data integration techniques and associated semantics.

The paper is organized as follows. In Section 2, we introduce the ESTEEM approach and the related peer architecture. A detailed description of the main components of the ESTEEM architecture is then provided in Sections 3, 4, 5, and 6. In Section 7, we show an example of P2P semantic cooperation in ESTEEM. Finally, related work and concluding remarks are discussed in Section 8 and 9, respectively.

2. THE ESTEEM APPROACH

In this section we highlight the ESTEEM approach to P2P semantic cooperation. In particular, we illustrate the structure of an ESTEEM peer in terms of its knowledge equipment and of its main functional components.

The goal of the ESTEEM approach is to support semantic cooperation among a set of autonomous and independent peers. To this end, ESTEEM relies on an overlay P2P network where i) *semantic communities* are defined to aggregate peers with similar interests and ii) a *probe/search mechanism* is adopted to enforce data and service discovery/sharing. An ESTEEM semantic community sc is defined as a pair of the form $sc = \langle CID, M \rangle$ where CID is the unique Community Identifier that characterizes the community sc and M is the Manifesto, that is the community ontology that describes the common interpretation (i.e., perspective) of the community interests. In ESTEEM, a semantic community is autonomously emerging, in that it originates from a proposal of a *community founder* (i.e., a peer) which initiates the community formation through dissemination of an advertisement message that contains CID and M of the emerging community. Each receiving peer p_i autonomously decides whether to join the community on the basis of its level of interest in the received manifesto M. Such a level of interest is computed by invoking an *ontology-based semantic matchmaker* and by evaluating the semantic affinity between M and the peer ontology of p_i. As shown in the example of Figure 1, an ESTEEM peer can join zero or more semantic communities according to the results of the semantic matchmaking process. Furthermore, communities are exploited as

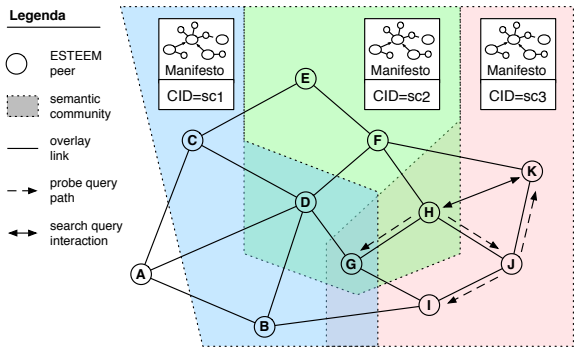

Figure 1: An ESTEEM P2P network

a *semantic overlay* on top of the basic P2P overlay (i.e., the *global overlay*) in order to enforce effective data and service sharing. In this respect, the probe/search mechanism is defined to distinguish:

- the **discovery phase**, based on ontology matching, where *probe queries* are defined to identify the peers that are capable of providing relevant knowledge with respect to a given topic of interest;
- the **sharing phase**, based on P2P mapping definition, where standard *search queries* are defined to point-to-point interact with a previously discovered peer for actual data acquisition and/or service invocation.

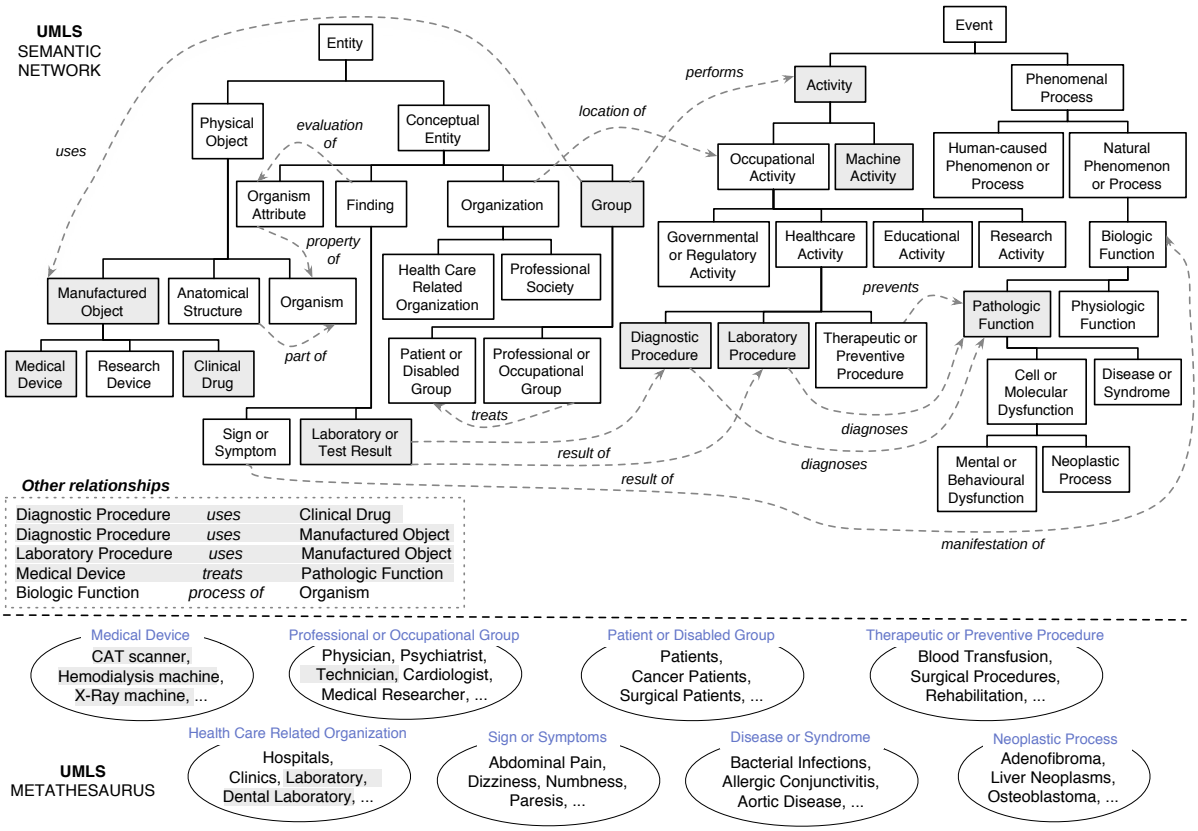

Figure 2: An example of Peer Ontology

In the discovery phase, the joined semantic communities are exploited by a requesting peer for selecting the probe query recipients with the aim of choosing those communities and peers that are most likely to provide relevant results according to the query target. In this context, the semantic matchmaker is invoked to evaluate the relevance of a community with respect to a probe query by comparing the community manifesto against the query content. In the example of Figure 1, the peer H submits a probe query and selects the community sc_3 as recipient since sc_3 is found to be relevant by the semantic matchmaker. By relying on the semantic overlay of sc_3 the probe query is received by the sc_3 members that hopefully reply to the requesting peer H with their matching knowledge. Collecting probe query replies, the peer H evaluates the results and decides whether to perform the sharing phase by directly interacting with the most interesting peers that provided a reply (i.e., peer K in the example) through appropriate search queries with the aim at accessing their data and services.

2.1 The knowledge equipment of an ESTEEM peer

The ESTEEM approach is characterized by the presence of a set of independent peers without prior reciprocal knowledge and no degree of relationship, that dynamically need to cooperate by sharing their resources (e.g., data, documents, services). Such a collaboration scenario is *multi-knowledge*, in that no centralized authorities are defined to manage a comprehensive view of the resources shared by all the nodes in the system, due to the high dynamism and variability of collaboration and sharing requirements. As a consequence, an ESTEEM peer joins the network by providing an ontology-based representation of the resources it intends to share with the other nodes of the system. In particular, an ESTEEM peer is characterized by a *peer ontology*, a *service ontology*, a *context dimension tree*, and a *data quality and trust profile*. Moreover, the manifesto of each joined semantic community is included in the knowledge equipment of an ESTEEM peer.

The peer ontology. The peer ontology is the core knowledge of a peer and provides a semantically rich description of the peer data that are available for sharing. In particular, the peer ontology is exploited during the discovery phase in order to evaluate whether matching knowledge can be returned to a requesting peer in reply to an incoming probe query. Furthermore, the peer ontology is also exploited for deriving the peer interests through connection with the context model of the peer and for determining the semantic communities to join. A peer defines its own ontology by acquisition from an external source or by composition through application of classical ontology engineering methodologies [20].

As an example according to the ESTEEM scenario, we consider the portion of peer ontology shown in Figure 2. This example is extracted from the Unified Medical Language System [1] and represents the knowledge of a peer belonging

[1] http://umlsinfo.nlm.nih.gov/

to the health care domain. In this example, we provide a graphical representation of the peer ontology that is characterized by a semantic network of concepts and semantic relations. Vocabularies about biomedical concepts are also included in the peer ontology in terms of a metathesaurus.

The service ontology. The service ontology provides a semantically rich description of the peer services that are available for sharing. Service descriptions represent functional aspects of a service, based on the WSDL standard for service representation, in terms of service category, service functionalities (operations) and input/output messages (parameters). In order to provide peer service descriptions, concepts in the peer ontology are used to express input and output messages (parameters) of services and constitute the so-called *Service Message Ontology (SMO)*. Furthermore, a *Service Functionality Ontology (SFO)* is defined for providing knowledge on the concepts used to express service functionalities (operations). Finally, peer services are organized in a service ontology with reference to the SMO and the SFO. In Figure 3, an example of SFO is given [2].

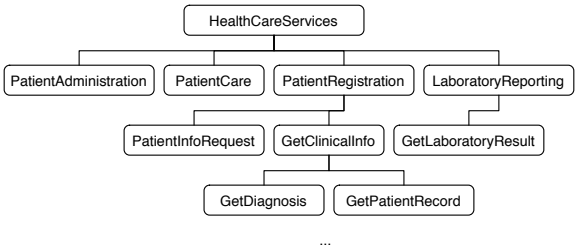

Figure 3: An example of Service Functionality Ontology

The context dimension tree. The *Context Dimension Tree (CDT)* [11] has been conceived to support the tailoring of the peer data according to the current context: our doctor, preparing for a stay in Central Africa, is interested in acquiring information on the diseases and symptoms common in that area, and on the available care facilities, possibly related to the correct period of the year. In another scenario, a lab technician in her working place needs/offers information and services related to the devices, procedures and analysis to be performed within the lab structure. Accordingly, the CDT of an application expresses the several perspectives determining what portion of data is interesting in the different situations. The user category, *actor*, the *situation* she may be in, the *interest topic* are some of the most commonly significant *dimensions*, driving the selection of relevant information/services. A dimension value can be further analyzed w.r.t. different viewpoints, generating further (sub)dimensions.

A subtree of the CDT, obtained by appropriately choosing a set of dimension values, is called a *context*, and determines a portion of the entire data set (a *data chunk*), specified as a view, representing the data that are relevant when the corresponding context becomes current. In Figure 4, we show an example of CDT modeling the possible contexts of our medical application. The peer CDT is defined by a CDT

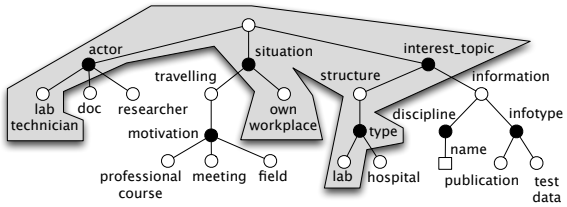

Figure 4: An example of Context Dimension Tree

designer according to the specific application. In our experience, the above dimensions are common to most applications, although it may happen that only some of them be needed, while more might be required. For example, many applications include the dimensions *space* which refers to the place where the user is currently located, and might be represented by GPS coordinates or by any other location information, and *time*, which is a temporal indication based on the current time. The CDT designer has the role to establish which dimensions are appropriate for the current scenario. Furthermore, the CDT designer specifies the correspondence between each given context of the CDT and the portion of the peer ontology (i.e., data chunk) that is relevant to it. As an example, the gray area of Figure 2 is the data chunk defined for the context represented by the gray part of Figure 4. A complete and formal definition of the CDT and its usage for data tailoring can be found in [10, 11]; here we have applied the model and adapted the methodology to the emergent semantic community scenario.

Data quality and trust profile. The data quality and trust profile involves the computation of data quality metrics on the peer data that are available to other peers. More specifically, each peer has the possibility of associating *quality metadata* to the exported data. Such metadata represent data quality measures corresponding to some specific quality dimensions. We have currently implemented metrics for those dimensions that are considered the most common among the ones defining data quality, namely: *column completeness*, *format consistency*, *accuracy* and *internal consistency* (see [5] for the definition of such metrics).

Once such quality metadata are available in the system, they can be used for evaluating the trust of a peer providing data to other peers of the community. When deciding the *atomic* unit to trust in an emergent semantics system, a first hypothesis could be to trust the peer as a whole, with respect to the totality of exchanged data or more generally to the transactions performed with other peers. The method proposed in [2] is an example of this case. Instead, we follow the approach of associating trust to a peer as a whole but we propose two major modifications: first, we consider a specific type of transaction, i.e. data exchanges; second, we evaluate trust of a peer with respect to a specific type of provided data. The key idea can be summarized as follows: (i) the atomic unit of trust, is the couple $\langle Peer_i, \mathcal{D} \rangle$, where $\mathcal{D}$ is an element of the peer ontology; (ii) the trust level of a peer P is computed on the basis of the number of complaints fired by other peers of the community, for

[2]The ARTEMIS Project: A Semantic Web Service-based P2P Infrastructure for the Interoperability of Medical Information Systems (http://www.srdc.metu.edu.tr/webpage/-projects/artemis/).

which P had been a data provider. The details of the model that we use for trust computation are provided in [16]. The major adaptation to the ESTEEM architecture is related to the consideration of each semantic community as a newly constituted cooperative information system, thus requiring a community specific trust computation service.

Semantic community manifestos. The peer stores the manifesto of each joined semantic community. We stress that a community manifesto is used to characterize the interests of the community participants and it is defined according to the preferences of the community founder (i.e., a peer). In general, the community manifesto is extracted from the peer ontology of the founder and it consists of a focused ontology. We note that the level of detail used for specifying the community manifesto depends on the community goal. In particular, the CDT can be used to support the user in specifying the community manifesto by allowing the founder to perform tailoring of the peer ontology. For instance, by using only the first level dimension nodes of the CDT, the founder selects the high-level concepts to specify the interests of the semantic community. Moreover, portions of the service ontology, the CDT, and the data quality and trust profile can be also included in the community manifesto to further specify the community objective.

2.2 The main components of the ESTEEM architecture

The ESTEEM architecture is defined to address the main requirements of a peer to support P2P semantic cooperation as described at the beginning of this section. As shown in Figure 5, the following main components are defined in the ESTEEM architecture to this end:

- *Network & overlay component.* It is responsible for managing the peer connectivity and for handling incoming and outgoing messages. From the network point of view, the ESTEEM P2P infrastructure is organized in semantic overlays featuring the semantic communities. In this respect, the network & overlay component is responsible for maintaining the overlays and the associated peer communications.
- *Semantic community & routing component.* It is responsible for managing the peer participation in semantic communities and for discovering the semantic neighborhood of a peer. Furthermore, this component is responsible for providing a semantic routing mechanism to effectively enforce query propagation.
- *Semantic matchmaking component.* It is responsible for providing semantic affinity evaluation when comparing different peer ontological descriptions. This component is invoked by a peer during the discovery phase to identify peers that are capable of providing matching resources (i.e., data, service, context) w.r.t. a given target request. Different techniques are provided by the semantic matchmaking component according to the type of matching resource that is specified in the request. In particular, ontology, service, and context matching techniques are provided by the semantic matchmaker.
- *Data & service discovery component.* It is responsible for interacting with the user and for satisfying its discovery requests. In particular, this component provides the functionalities for context and quality/trust management. Furthermore, discovery and sharing functionalities are also addressed in this component through query/answer and P2P mapping management.

In the following sections, a detailed description of the ESTEEM architecture components is provided.

3. NETWORK & OVERLAY

The network & overlay component manages the peer connectivity and the participation in the semantic overlays. Three modules are defined in the network & overlay component, that is the *global overlay*, the *semantic overlay*, and the *preferential link* modules.

Global overlay. The global overlay module aims at maintaining connected the general overlay network, called Global Overlay (GO), in order to enforce communication among peers. The GO is a logical network collecting all the peers participating the system: each node represents a peer and each link is a logical connection between two peers. In order to guarantee connection of GO, an Overlay Management Protocol (OMP) is used, which defines some specific procedures to join, leave and modify the GO. In ESTEEM, a shuffling-based OMP is chosen in order to allow more effective information diffusion among peers [33]. This kind of OMP arranges the GO as a graph in which each peer is directly connected to a very small portion of the entire peer population, and it is also transitively connected with all other peers through redundant short paths.

On top of the GO, semantic community manifestos are continuously circulating in order to allow peers to discover the existing communities and to eventually join one (or more) of them. Each peer maintains a table, called Semantic Overlay Table (SO_Table) where it stores information about the discovered existing communities. Each entry of the table is represented by the tuple $\langle CID, M, N\_ap \rangle$ where CID and M are the community identifier and associated manifesto, respectively, and N_ap is the peer which acts as access point for that community.

When receiving the manifesto of an existing community, the choice to join it is performed by invoking the community membership module where ontology matching techniques are used to evaluate the peer's level of interest in the community. In case that a high level of interest in the community is returned by the community membership module, the peer is "promoted" to the Semantic Overlay Layer joining the corresponding semantic community. When the existing communities are not satisfying for the peer, a new semantic community can be founded by defining the CID and the related manifesto to be advertised in the GO. We note that this strategy implies that a peer may not find a community of interest even if it exists. For this reason, a mechanism for merging similar semantic overlay merging is defined in ESTEEM. In particular, such mechanism relies on ontology matchmaking techniques for enabling a peer to detect semantically related manifestos and to promote a merged community where the manifesto is specified as evolution of the former ones. A detailed description of merging and evolution techniques for P2P semantic overlays are provided in [4].

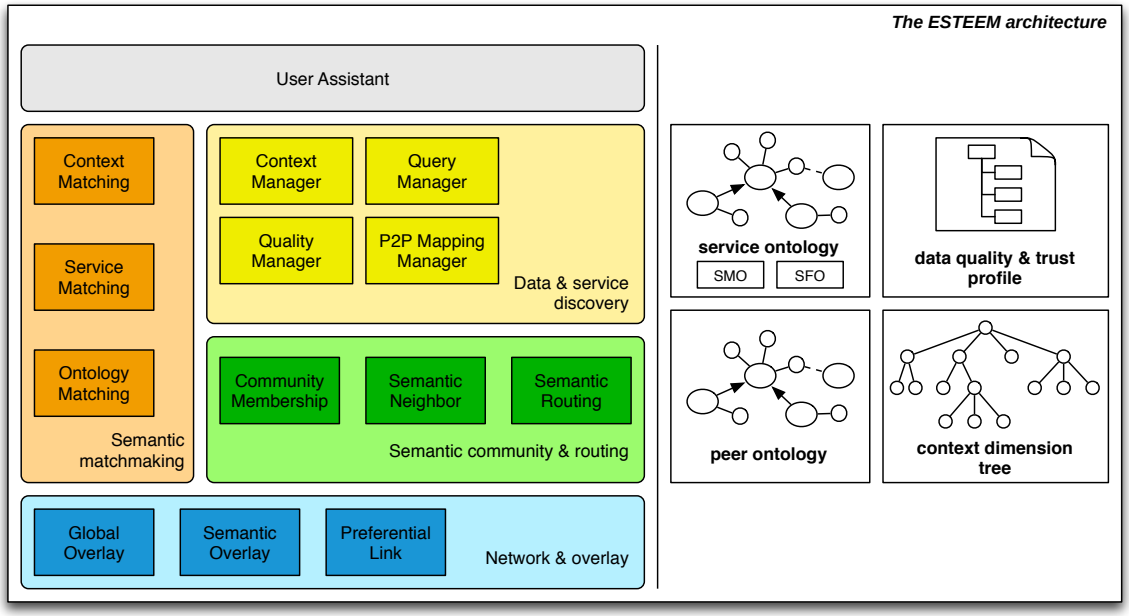

Figure 5: The reference architecture of an ESTEEM peer

Semantic Overlay. Similarly to the global overlay module, the semantic overlay module arranges nodes in Semantic Overlays, i.e. logical networks in which peers sharing the same interests are clustered together. Also in this case, an OMP is used to maintain connectivity; this OMP is the same one used at the global overlay level, with the difference that here all the procedures work with a subset of the network peers.

The Semantic Overlay Component is also responsible for communication at the semantic community level and in particular it is responsible for the dissemination of the probe queries.

Preferential Link. This module manages the creation of preferential links. A preferential link is a preferred connection established with a peer more semantically related for a particular aspect. Preferential links can be set using different criteria like the affinity on desired data or services. The method used to select a preferred neighbor is a probe query, that is a query spread inside the semantic overlay to understand which are the more interesting peers to interact with. We stress that preferential links are temporary links set for a specific interaction and then replaced at the next one.

4. SEMANTIC COMMUNITY & ROUTING

The semantic community & routing component has the responsibility to discover and maintain the *semantic neighborhood* of a peer in order to link those communities and those single peers that share similar contents. To this end, the semantic community & routing component includes the *community membership*, the *semantic neighbor*, and the *semantic routing* modules.

Community membership. The community membership module has the responsibility to manage the peer participation in semantic communities. On one side, the goal of this module is to evaluate the level of peer interest in the existing communities and to join the most interesting ones. When the advertisement of a community sc is received, the ontology matchmaking module is invoked to match the associated manifesto M against the peer ontology and to identify possible semantic affinities. The matching results are used to evaluate the peer interest in sc (i.e., the higher the matching results, the higher the peer interest in the community) and to decide whether to join it. The manifesto of each joined community is stored by the peer and exploited by the semantic routing module for supporting query propagation on a semantic basis. On the other side, the goal of the community membership module is to promote the formation of a new

semantic community when existing communities are not sufficiently interesting for the peer. In this case, the peer acts as community founder and has the role to define a new manifesto and to advertise it to the other network nodes. In both cases, the community membership module mainly interacts with the global overlay module to receive/disseminate community manifestos from/to the other network peers.

Semantic neighbor. The semantic neighbor module has the responsibility to discover the single peers that are capable of providing matching resources in response to a given target and to link them as semantic neighbors. The goal of this module is to observe the results collected through past probe queries and to use them for learning the contents of the other peers. In particular, the semantic neighbor module allows to link a concept c in the peer ontology (or a service s in the service ontology) with the network peers (i.e., semantic neighbors) that can provide matching resources for c. Furthermore, a semantic neighbor p is associated with a *confidence value* which describes the expertise level of p in providing matching replies for c. Semantic neighbors change during time due to join/leave operations of peers and to modifications of peer context and interests. Probe query replies are used to progressively detect neighbor changes and to update confidence values accordingly. The semantic neighbor module mainly interacts with the semantic overlay module for establishing point-to-point connections with selected semantic neighbors and for supporting the sharing phase. Moreover, confidence values can be exploited to set preferential links with the peers with the highest expertise on most interesting topics. A detailed description of the ESTEEM techniques for computing confidence values is provided in [15].

Semantic routing. The semantic routing module has the responsibility to select the query recipients on a semantic basis, by identifying those peers that are most likely to provide matching results according to the query target. The goal of this module is to compose the recipient list of a query by exploiting i) the joined semantic communities stored by the community membership module and ii) the semantic neighbors stored by the semantic neighbor module. Given a query to be submitted to the network, the semantic routing module implements a semantic-based query propagation mechanism which allows to identify the recipients, either semantic communities or semantic neighbors, having the highest semantic affinity with the query target. In this respect, the ontology matchmaking module is invoked to evaluate the level of semantic affinity between the query target and the contents of each potential recipient. In particular, the semantic matchmaker compares the query target against the manifesto of each joined community and against the discovered contents of semantic neighbors. Confidence values are then exploited to rank semantic neighbors according to their relevance for the query target. Finally, matching communities and top-ranked semantic neighbors are selected as query recipients. A detailed description of the ESTEEM semantic routing techniques is provided in [15].

5. SEMANTIC MATCHMAKING

The semantic matchmaking component provides semantic affinity evaluation when comparing different peer ontological descriptions. In ESTEEM, semantic matchmaking is required either for performing peer ontology matching, or for service ontology matching, or for context matching. For this reason, the semantic matchmaking component includes the *ontology matching*, the *service matching*, and the *context matching* modules.

Ontology matching. The HMatch ontology matchmaking system [13] is exploited in ESTEEM for the selection of the semantic communities to join and for supporting the identification of semantic neighbors. The choice of HMatch is motivated by the fact that it has been specifically conceived to work in open environments where flexibility and dynamic configurability are essential requirements. HMatch performs ontology matching at different levels of depth by deploying four different *matching models* spanning from surface to intensive matching, with the goal of providing a wide spectrum of metrics suited for dealing with many different matching scenarios that can be encountered in comparing concept descriptions of real ontologies. HMatch takes two ontologies as input and returns the mappings that identify corresponding concepts in the two ontologies, namely the concepts with the same or the closest intended meaning. A threshold-based mechanism is enforced to set the minimum level of semantic affinity required to consider two concepts as matching concepts. Given two concepts c and c', HMatch calculates a semantic affinity value $SA(c, c') \in [0, 1]$ as the linear combination of a linguistic affinity value $LA(c, c')$ and a structural affinity value $TA(c, c')$. The HMatch linguistic affinity provides a measure of similarity between two ontology concepts c and c' computed on the basis of their linguistic features (i.e., concept names). For the linguistic affinity evaluation, HMatch relies on a thesaurus of terms and terminological relationships automatically extracted from the WordNet lexical system. The HMatch structural affinity provides a measure of similarity by taking into account the structural features of the ontology concepts c and c'. In HMatch, the structure of a concept can include properties, semantic relations with other concepts, and property values. Moreover, four matching models, namely *surface*, *shallow*, *deep*, and *intensive*, are defined to allow a flexible composition of the concept structure according to the level of semantic complexity that is considered. In the surface matching, only the linguistic affinity between the concept names of c and c' is considered to determine concept similarity. In the shallow, deep, and intensive matching, also structural affinity is taken into account to determine concept similarity. In particular, the shallow matching computes the structural affinity by considering the structure of c and c' as composed only by their properties. Deep and intensive matching extend the depth of concept structure by considering also semantic relations with other concepts (deep matching model) as well as property range (intensive matching model), respectively. A comprehensive semantic affinity value $SA(c, c')$ is evaluated as the weighted sum of the linguistic affinity value and the structural affinity value, that is $SA(c, c') = W_{LA} \cdot LA(c, c') + (1 - W_{LA}) \cdot TA(c, c')$ where $W_{LA} \in [0, 1]$ is a weight expressing the relevance assigned to the linguistic affinity in the semantic affinity evaluation process. A *matching policy* MP is defined in HMatch to configure the current execution of the matchmaker for a given matching case. A matching policy is a triple of the form $MP = \langle mm, W_{LA}, t \rangle$, where: $mm \in \{$surface, shallow, deep, intensive$\}$ denotes the matching model to be used for

HMatch execution, $W_{LA} \in [0,1]$ denotes the linguistic affinity weight, and $t \in (0,1]$ denotes the matching threshold.

Service matching. Service matchmaking is performed by comparing service descriptions, combining together different matching models: (i) a deductive model, exploiting deduction algorithms for analyzing service descriptions [8], (ii) a similarity-based model, where retrieval metrics are applied to measure the degree of match between services [9].

The deductive approach is applied with the support of a DL reasoner to classify the match between a service request $\mathcal{R}$ and a service advertisement $\mathcal{S}$ on the basis of the ontological knowledge. Classification is made according to the following kinds of matches:

- *Exact match*, to denote that $\mathcal{S}$ and $\mathcal{R}$ have the same capabilities, that is, they have: (i) equivalent operations; (ii) equivalent output parameters; (iii) equivalent input parameters;
- *Plug-in match*, to denote that $\mathcal{S}$ offers at least the same capabilities of $\mathcal{R}$, that is, names of the operations in $\mathcal{R}$ can be mapped into operations of $\mathcal{S}$ and, in particular, the names of corresponding operations, input parameters and output parameters are in any generalization hierarchy in the peer ontology; the inverse kind of match is denoted as subsume;
- *Intersection match*, to denote that $\mathcal{S}$ and $\mathcal{R}$ have some common operations and some common I/O parameters, that is, some pairs of operations and some pairs of parameters, respectively, are related in any generalization hierarchy in the peer ontology.
- *Mismatch*, otherwise.

Similarity analysis is applied to quantify the match between services. In particular, when exact match occurs, similarity between services is set to 1 (full similarity). Otherwise, when plug-in/subsume and intersection match occur, similarity coefficients are computed to further refine in a quantitative way the ranking of returned services. Finally, when mismatch occurs, the similarity value is set to zero. The similarity coefficients are based on the Dice's metrics and have been widely experimented. A detailed description of the similarity coefficients and their application is given in [9].

The semantic matchmaking techniques identify semantic links between services maintained in the service ontology. In particular, a semantic link between two services is established if the kind of match is not mismatch and the similarity value is equal or greater than a given threshold. A semantic link is identified by the kind of match and it is weighted by the similarity value expressed by the similarity coefficient. Semantic links are expressed in the service ontology and can be established between services belonging to the same peer or between services belonging to different peers.

Context matching. In context matchmaking, the goal is to compare the concepts (i.e., nodes) belonging to different CDTs and to identify the possible correspondences. To this end, traditional string-based matching technique can be adopted. Moreover, some peculiar aspects of context matchmaking need to be considered. In particular, concepts need to be compared w.r.t. the types (black or white) of the possibly matching nodes. Two contexts, i.e. two subtrees of the same CDT might be equal, incomparable, or one strictly contained into the other. The first case is obviously the easy one, since there is full correspondence of contexts. Also the containment case is easily dealt with, because in this case the more general context is chosen as the common one, an affinity value 1 is returned, and all further exchanges may be performed on the basis of the more general context. In the case of incomparable subtrees, an affinity value is computed, based on the global affinity of the data chunks associated with the two contexts under analysis.

6. DATA & SERVICE DISCOVERY

This component comprises various modules providing the peer with different functionalities to be exploited at query time for enforcing the discovery of both data and services.

Context manager. It is responsible for managing the peer CDT. In ESTEEM, context is used at different times, in the initial phase, when the semantic community is created, as well as later on, during its life-cycle. The context manager is used to support the peer in the following tasks:

- *CDT mapping definition.* For each possible context specified by the CDT, the context manager supports the user in defining mappings between the CDT and the associated portions of peer ontology. Such mappings are then exploited for probe query answering, when a peer uses context similarity to look for other peers providing the data and/or services it needs.
- *Context-based query formulation.* The peer context is exploited to support the user in formulating queries. In particular, two different approaches can be distinguished. On one side, the peer context can be used to directly formulate a query when a peer is interested in discovering nodes using a similar context. In this case, the query contains a context the peer is interested in (i.e. a subtree of the CDT, containing one or more dimension values). On the other side, the peer context can be used to indirectly formulate a query by supporting the user in specifying the concepts of interest to insert in the query. In this case, the user selects the context of interest from the peer CDT. Mappings between CDT and peer ontology are then exploited to define the query contents according to the peer ontology concepts associated to the context of interest.
- *Data caching.* In ESTEEM, a peer may need to collect data to be used later on, when the network connection to the joined semantic communities will not be available. This may happen, for instance, when a small device (e.g., palm computer, smartphone) is considered, or for caching purposes. As a consequence, the peer prepares such data according to an upcoming context, by tailoring the portion of interesting data for such context. The tailored information can be extracted from the local peer data or be acquired from external peers through the probe/search mechanism provided in ESTEEM when the connection is available.

Quality manager. It is invoked during query processing in order to exploit quality metadata and to take data inconsistencies into account. More specifically, in ESTEEM, we

assume that data can exhibit key-level conflicts [6]. This implies that an object identification step must be performed in order to provide answers to user queries. Due to the specific requirements of the ESTEEM system, this step should be performed in a fully automatic way. In particular, when a record linkage process is enacted, it is often the case that a reduction of the space of the possible matching records must be performed. This phase involves the choice of one or more attributes, referred to as matching keys, that is normally performed by humans. In our context, we need that this phase be also performed automatically. Therefore, we have added to the metadata calculated for quality profiling, a further metadata, named *identification power*, that specifies how much a given attribute is discriminating when trying to match objects. For instance a *Sex* attribute is quite surely more discriminating than a *Surname* attribute, when matching records referred to persons. We adopt an automatic method for matching key computation which is based on the identification power and on quality metadata; the method is fully described in [7]. Once automatized this phase, we are able to run an object identification process with the objective of solving key-level conflicts, thus allowing query processing to being carried out. With reference to the nature of the record linkage process, we are currently testing the appropriateness of deterministic versus probabilistic linkage in the ESTEEM architecture context [19].

Query manager. It is responsible for query composition and answer management. During the discovery phase, the query manager is invoked for supporting the user in probe query formulation. In this phase, interactions with the context manager can be performed to define probe queries by exploiting the peer context. Furthermore, the query manager is invoked during the sharing phase to allow the formulation of search queries and to perform data acquisition and/or service invocation according to the results of the discovery phase. In this respect, interactions with the P2P mapping manager are executed to establish mappings with the peer ontology of a selected peer (i.e., *sharing partner*). Moreover, interactions with the quality manager are also performed to include quality and trust requirements in queries before their submission to the network. A detailed description of the probe/search approach for resource discovery and sharing in P2P systems is provided in [14].

P2P mapping manager. It is responsible of establishing a point-to-point communication between two peers for performing the sharing phase. The P2P mapping manager is invoked by a requesting peer to access the resources (i.e., data and services) provided by a peer discovered as a semantic neighbor during the probe phase. The goal of the P2P mapping manager is to define appropriate mappings between the involved peer ontologies in order to enforce query answering and service invocation. In ESTEEM, the PIAZZA approach for P2P mapping definition is currently adopted [21]. In particular, two types of mappings, namely *peer* and *definitional* mappings, are defined. Peer mappings describe the correspondences between the data stored in two different peers, while definitional mappings define the correspondences between the terminology and structure of two peer ontologies (or the correspondences between operations and I/O parameters in two service ontologies). Such an approach is alternative to the one used by the Hyperion project [24]. Hyperion mappings rely on mapping tables that store the correspondence between values. Such tables are often the result of expert knowledge and are manually created by domain specialists. However, mechanisms to partially support automatic mapping discovery can be used: new mappings can be inferred from already existing ones. We choose the PIAZZA approach as it is indeed able to use traditional Global-As-View (GAV) and Local-as-View (LAV) mappings to describe the semantic relationships between peers.

7. P2P SEMANTIC COOPERATION IN ESTEEM

According to the architecture shown in Figure 5, we now outline how ESTEEM can be exploited for P2P semantic cooperation.

7.1 Joining the ESTEEM network

In order to join the ESTEEM network, a peer invokes the overlay management protocol, and gets inserted in the global overlay with a unique identifier. Once the peer becomes part of the global overlay, it searches for semantic communities that match its interests. To this end, the SO_Table maintained by the peer is exploited to evaluate the level of interest in the acquired community manifestos. Furthermore, a *search procedure* is invoked to look for additional manifestos when interesting communities are not found in the SO_Table. According to the matching results, interesting communities are joined, otherwise a new community is proposed by the peer. The joined communities are then exploited by the peer for data/service discovery and sharing.

7.2 Community-based data discovery

Once the semantic communities of interest are joined, the ESTEEM network can be queried by a peer according to the probe/search approach illustrated in Section 2. To this end, the query manager is invoked to formulate a probe query containing a target request. As described in Section 6, the context manager and the quality manager can be involved in supporting query formulation. The semantic routing module is then invoked to select as query recipients (i.e., semantic communities and semantic neighbors) the peers with more chances to provide matching results. Finally, the probe query is submitted to the ESTEEM network through the network & overlay component. When the goal is data discovery, the query contains one or more concepts of interest extracted from the peer ontology of the requesting peer. Receiving a probe query, a peer invokes the semantic matchmaker to compare the request against the concepts contained in its peer ontology. As a result, the matching concepts returned by the matchmaker are handed back to the requesting peer. Furthermore, the CDT of a peer can be used to specify a probe query for identifying those peers that share a similar context. In this case, the probe query contains a target context (i.e. a subtree of the CDT) and a receiving peer can use context matching techniques to determine whether its context matches the request. When a positive matching is found, a reply is sent to the requesting peer together with the discovered matching context. Probe replies are considered by the requesting peer to i) discover new semantic neighbors (see Section 4) and ii) to select most interesting peer for interaction in the sharing phase.

Example. As an example of community-based data discovery, we consider the portion of ESTEEM network shown in Figure 1. In this example, the probe query is received by the sc_3 members and their semantic matchmaker is used to evaluate whether they are capable of replying to the requesting peer H with matching knowledge. Two cases are considered:

i) The probe query contains concepts of the peer ontology. Suppose, for example, that peer H's probe query contains the concept Lab_procedure, and that two answers are provided, by peer K and peer J. In particular, the peer K contains the concept Lab_protocol, with an affinity value SA(Lab_procedure, Lab_protocol) = 0.8, while the peer J contains the concept Diagnostic_protocol, with an affinity value SA(Lab_procedure, Diagnostic_protocol) = 0.4. After the probe phase, the peer H decides to further interact only with the peer K by asking for its data schema, in order to proceed with the appropriate search query.

ii) The probe query contains a context of interest. In this case, suppose that peer H sends a probe query for the context of Figure 4, and receives two answers from the peer K and the peer J. In particular, the peer K contains the same context (context affinity value equal to 1), while the second one contains a slightly different one, namely, where the actor is a Doctor instead of a Lab_technician, with an affinity value 0.6. After the probe phase, peer H decides to further interact only with peer K, and asks Peer K for the schema of the data corresponding to the exchanged context, in order to proceed with the appropriate search query.

7.3 Community-based service discovery

For service discovery, the probe query contains the description of the service interface in terms of required service functionalities, each of them described through the operation name and names of input/output parameters. When a peer receives a probe service request, it matches the service request against its own service descriptions by applying the service matchmaking techniques explained in Section 5. Then, it sends back to the requesting peer a message containing the kind of match and the similarity degree between the probe service request and its own services. Replies to probe service requests are used by the requesting peer to build a map of its semantic neighbors, with the similar services and the semantic relationships with them. After the discovery phase, a service request $\mathcal{S}_\mathcal{R}$ can be specified by a peer p and a list of candidate services $CS = \{\langle \mathcal{S}_1, GSim_1, mt_1\rangle, \ldots \langle \mathcal{S}_n, GSim_n, mt_n\rangle\}$ is retrieved according to the results collected with probe query replies, where $\mathcal{S}_i$ is a candidate service with corresponding similarity values $GSim_i$ and match type mt_i.

If a service $\mathcal{S}_i$ presents an exact or a plug-in match with the request, then $\mathcal{S}_i$ satisfies completely the required functionalities and it is not necessary to forward the service request to the semantic neighbors. Otherwise, if a service $\mathcal{S}_i$ presents a subsume or an intersection match with the request, the peer p forwards the request to those peers that are semantic neighbors of p with respect to $\mathcal{S}_i$. Note that p does not consider semantic neighbors that presents a subsume or an exact match with $\mathcal{S}_i$, because this means that they provide services with the same functionalities or a subset of the functionalities of $\mathcal{S}_i$ and they cannot add further capabilities to those already provided by $\mathcal{S}_i$ on the peer p. A list of semantic neighbors $SN = \{\langle p_1, \{\mathcal{S}_1, GSim_1, mt_1, \ldots \mathcal{S}_{m_1}, GSim_{m_1}, mt_{m_1}\}\rangle, \ldots \langle p_k, \{\mathcal{S}_1, GSim_1, mt_1, \ldots \mathcal{S}_{m_k}, GSim_{m_k}, mt_{m_k}\}\rangle\}$ is obtained in this phase and is used to forward the original request.

The semantic neighbors in SN can be ranked with respect to their relevance with the original request. Given a semantic neighbor $sn \in SN$, its relevance with respect to the request $\mathcal{S}_\mathcal{R}$ is computed according to the following equation:

$$r_{sn} = \frac{\sum_{i=1}^{m_j} \frac{2 * GSim_i * GSim(\mathcal{S}_\mathcal{R}, \mathcal{S}_i)}{GSim_i + GSim(\mathcal{S}_\mathcal{R}, \mathcal{S}_i)}}{m_j} \quad (1)$$

Ranking of semantic neighbors is exploited to constrain the forwarding according to a threshold-based mechanism.

Example. Let consider a peer A providing three services GetDesease, GetDiagnosis and GetLaboratoryResult for which the following semantic neighbors have been found (see Figure 6):

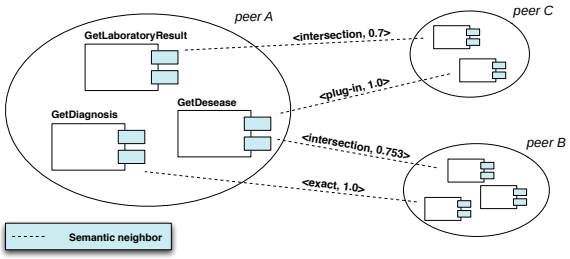

Figure 6: An example of semantic neighbors for the peer A

GetDesease	⟨ peer B, 0.753, intersection ⟩
	⟨ peer C, 1.0, plug-in ⟩
GetDiagnosis	⟨ peer B, 1.0, exact ⟩
GetLaboratoryResult	⟨ peer C, 0.7, intersection ⟩

Let suppose that for a given request $\mathcal{S}_\mathcal{R}$ on the peer A, we obtain the following list of candidate services: CS = {⟨GetDesease,0.9,intersection⟩, ⟨GetDiagnosis,0.7,subsume⟩}. Obviously, {⟨GetLaboratoryResult,0.0,mismatch⟩} is excluded from CS. For what concerns GetDesease, both peer B and peer C must be considered as candidate semantic neighbors, since they could provide some additional capabilities with respect to peer A. Moreover, for what concerns GetDiagnosis, peer C is not a semantic neighbor, while peer B has a related service, that presents an exact match with GetDiagnosis. This means that peer B has no additional capabilities to offer with respect to those already provided by GetDiagnosis in peer A: SN = {⟨ peer C, {GetDesease,1.0, plug-in} ⟩, ⟨ peer B, {GetDesease, 0.753,intersection} ⟩}. According to (1), the ranking values for peer B and peer C with respect to the request $\mathcal{S}_\mathcal{R}$ are $r_{\text{peer B}} = 0.726$ and $r_{\text{peer C}} = 0.947$, respectively.

8. RELATED WORK

Recent research work on P2P systems aims at evolving from basic P2P networks, supporting only file exchanges with simple filenames as metadata, to systems based on rich metadata like ontologies, supporting the exchange of structured contents [21, 22]. At the network level, the recent growth of P2P applications has motivated the interest in general-purpose P2P overlay structures. In this respect, a

challenging issue is to support complex applications without overloading network resources and by maintaining scalability at the same time. To this end, techniques for semantic query propagation are being proposed to select query recipients by exploiting the replies to previous requests [30, 34]. Moreover, techniques for supporting the formation of semantic communities of peers are also being developed as a solution for improving the current level of cooperation in P2P systems. For instance, in [12] the KEx platform (Knowledge Exchange system) is defined for supporting peer federations where knowledge is organized from an individual- or community-based perspective and different meanings are managed through a semantic matching algorithm.

A basic requirement in P2P community management and resource sharing is related to security and trust, to allow a peer to retain ownership of its own contents and processing capabilities, and to allow other peers to access them under appropriate conditions. Various works investigated mechanisms to establish consensus on trust, taking into account reputation of referrals [27]. In P2P systems, trust issues are also related to the quality of data exported by information producers. The quality of information is either dependent upon people opinions or upon the applications consuming that information. In the former case, consensus computation will be a crucial component in determining the quality of information [16].

Most semantic approaches to data/service discovery are based on ontologies and ontology matching techniques. The general goal of ontology matching is to compare different ontological descriptions for finding concepts that have a semantic affinity. A survey of ontology matching techniques is provided in [17], where formal and heuristic approaches are classified. Specific techniques for addressing ontology matching in open distributed environments, like P2P systems, are also presented in [13]. Furthermore, modern approaches for semantic matching between service descriptions are available to support service discovery in highly variable environments [28, 35].

In open distributed systems, data integration issues have been recently addressed [21]. In particular, no shared ontology is available to provide an explicit semantics to data. Initial approaches rely on a pre-defined corpus of terms serving as an initial context for defining new concepts [32] or make use of gossiping and local translation mappings to incrementally foster interoperability in the large [1]. Despite an initial formalization (see for instance [26]), there is still a fundamental lack of understanding behind the basic issues of data integration in P2P systems and initial proposals make unrealistic restrictions on the overall topology of the P2P mappings [18]. The problem still awaits proper characterization for real P2P systems, where mappings may have an arbitrary structure, possibly involving cycles. In this context, query processing is still an open issue.

Much research is currently devoted to context-aware, ubiquitous distributed systems, leading to the proposal of a variety of context models; although a lot of work has been done, the representation and management of the context can hardly be considered to be an assessed issue. Interesting surveys on context-aware systems and models have been proposed, for example [3, 31]. However, most of this research proposes the use of context for other purposes than data tailoring: some stress the problem of content presentation and adaptation to the delivery channel, others focus on location and situation management, others on modeling the user's activity, and still others work on context agreement and sharing.

With respect to the previous approaches, the original and distinguishing contribution of ESTEEM can be summarized as follows: i) the construction of a semantic overlay network by means of a shuffling-based mechanism combined with semantic matching results, ii) the definition of a probe query approach for both data and service discovery in P2P systems, and iii) the capability of incorporating information about the user context and the trust & quality of data into the query formulation and thus into the knowledge discovery process.

9. CONCLUDING REMARKS

In this paper, we have presented the ESTEEM approach to semantic cooperation in dynamic and multi-knowledge environments. A key feature of ESTEEM is to preserve the autonomous and spontaneous nature of peer communities while offering peers a rigorous and powerful approach to data/service discovery and sharing. In particular, non-obvious and important aspects, such as context, quality and trust, are also taken into account while building semantic cooperation. Ongoing work has to do with the implementation and testing of the various architecture components. A mock-up of user interface has also been sketched in order to run a first user trial related with the medical scenario illustrated in the paper. The trial will involve some medical doctors who already participated to the first collection phase of the ESTEEM requirements.

Acknowledgements. The ESTEEM architecture is the result of a team effort. For this reason, we have chosen to attribute the paper to a pen name. The following persons contribute to the ESTEEM project with their work. Carola Aiello, Roberto Baldoni, Devis Bianchini, Cristiana Bolchini, Silvia Bonomi, Silvana Castano, Tiziana Catarci, Carlo A. Curino, Valeria De Antonellis, Alfio Ferrara, Michele Melchiori, Diego Milano, Stefano Montanelli, Giorgio Orsi, Antonella Poggi, Leonardo Querzoni, Elisa Quintarelli, Rosalba Rossato, Denise Salvi, Monica Scannapieco, Fabio A. Schreiber, Letizia Tanca, Sara Tucci Pergiovanni.
In particular, the contribution of Stefano Montanelli in the form of valuable coordination and editorial support was essential for composing this paper.

10. REFERENCES

[1] K. Aberer, P. Cudrè-Mauroux, and M. Hauswirth. The Chatty Web: Emergent Semantics through Gossiping. In *Proc. of the 12th Int. World Wide Web Conference (WWW 2003)*, Budapest, Hungary, 2003.

[2] K. Aberer and Z. Despotovic. Managing Trust in a Peer-2-Peer Information System. In *Proc. of the 10th Int. Conference on Information and Knowledge Management*, Atlanta, Georgia, USA, 2001.

[3] M. Baldauf, S. Dustdar, and F. Rosenberg. A Survey on Context-Aware systems. *Int. Journal of Ad Hoc and Ubiquitous Computing*, 2007. To appear.

[4] R. Baldoni, R. Beraldi, V. Quema, L. Querzoni, and S. T. Piergiovanni. TERA: Topic-based Event Routing for Peer-to-Peer Architectures. In *Proc. of the 1st Int.*

Conference on Distributed Event-Based Systems (DEBS 2007), Toronto, Canada, 2007.

[5] C. Batini and M. Scannapieco, editors. *Data Quality: Concepts, Methods, and Techniques (Chapther 2).* Springer, 2006.

[6] C. Batini and M. Scannapieco, editors. *Data Quality: Concepts, Methods, and Techniques (Chapther 5).* Springer, 2006.

[7] P. Bertolazzi, L. De Santis, and M. Scannapieco. Automatic Record Matching in Cooperative Information Systems. In *Proc. of the ICDT Int. Workshop on Data Quality in Cooperative Information Systems (DQCIS'03)*, Siena, Italy, 2003.

[8] D. Bianchini, V. De Antonellis, M. Melchiori, and D. Salvi. Semantic-enriched Service Discovery. In *Proc. of the IEEE ICDE Int. Workshop on Challenges in Web Information Retrieval and Integration (WIRI 2006)*, Atlanta, Georgia, USA, 2006.

[9] D. Bianchini, V. De Antonellis, B. Pernici, and P. Plebani. Ontology-based Methodology for *e*-Service discovery. *Journal of Information Systems, Special Issue on Semantic Web and Web Services*, 31(4-5), 2006.

[10] C. Bolchini, F. A. Schreiber, and L. Tanca. A Methodology for Very Small DataBase Design. *Information Systems*, 32(1), 2007.

[11] C. Bolchini, C. Curino, E. Quintarelli, F. A. Schreiber, and L. Tanca. Context Information for Knowledge Reshaping. *Int. Journal on Web Engineering and Technology*, 2007. To appear.

[12] M. Bonifacio, P. Bouquet, G. Mameli, and M. Nori. Peer - Mediated Distributed Knowledge Management. In *Proc. of Int. Symposium on Agent Mediated Knowledge Management*, Stanford, CA, USA, 2003.

[13] S. Castano, A. Ferrara, and S. Montanelli. Matching Ontologies in Open Networked Systems: Techniques and Applications. *Journal on Data Semantics (JoDS)*, V, 2006.

[14] S. Castano, A. Ferrara, and S. Montanelli. *Web Semantics and Ontology*, chapter Dynamic Knowledge Discovery in Open, Distributed and Multi-Ontology Systems: Techniques and Applications. Idea Group, 2006.

[15] S. Castano and S. Montanelli. Semantically Routing Queries in Peer-based Systems: the H-Link Approach. *The Knowledge Engineering Review*, 2007. To appear.

[16] L. De Santis, M. Scannapieco, and T. Catarci. Trusting Data Quality in Cooperative Information Systems. In *Proc. of the 11th Int. Conference on Cooperative Information Systems (CoopIS'03)*, Catania, Italy, 2003.

[17] N. F. Noy. Semantic Integration: a Survey of Ontology-based Approaches. *SIGMOD Record Special Issue on Semantic Integration*, December 2004.

[18] R. Fagin, P. G. Kolaitis, R. J. Miller, and L. Popa. Data Exchange: Semantics and Query Answering. *Theoretical Computer Science*, 336(1):89–124, 2005.

[19] M. Fortini, M. Scannapieco, T. Tosco, and T. Tuoto. Towards an Open Source Toolkit for Building Record Linkage Workflows. In *Proc. of the SIGMOD Workshop on Information Quality in Information Systems (IQIS'06)*, Chicago, USA, 2006.

[20] A. Gomez-Perez, M. Fernandez-Lopez, and O. Corcho. *Ontological Engineering*. Springer Verlag, 2003.

[21] A. Halevy, Z. Ives, J. Madhavan, P. Mork, D. Suciu, and I. Tatarinov. The Piazza Peer Data Management System. *IEEE Transactions on Knowledge and Data Engineering*, 16(7):787–798, 2004.

[22] J. Broekstra et al. A Metadata Model for Semantics-Based Peer-to-Peer Systems. In *Proc. of the 1st WWW Int. Workshop on Semantics in Peer-to-Peer and Grid Computing (SemPGRID 2003)*, Budapest, Hungary, 2003.

[23] K. Aberer et al. Emergent Semantics Principles and Issues. In *Proc. of the 9th Int. Conference on Database Systems for Advances Applications - DASFAA 2004*, Jeju Island, Korea, 2004.

[24] A. Kementsietsidis, M. Arenas, and R. J. Miller. Mapping Data in Peer-to-Peer Systems: Semantics and Algorithmic Issues. In *Proc. of the ACM SIGMOD Int. Conference on Management of Data*, San Diego, California, USA, 2003.

[25] M. Lenzerini. Data Integration: A Theoretical Perspective. In *Proc. of the 21st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS 2002)*, Madison, Winsconsin, USA, 2002. Invited tutorial.

[26] M. Lenzerini. Principles of P2P Data Integration. In *Proc. of the 3rd Int. CAiSE Workshop on Data Integration over the Web*, Riga, Latvia, 2004.

[27] S. Marti and H. Garcia-Molina. Limited Reputation Sharing in P2P Systems. In *Proc. of the 5th ACM Conference on Electronic Commerce ACM-EC 2004*, New York, NY, USA, 2004.

[28] A. Patil, S. Oundhakar, A. Sheth, and K. Verma. METEOR-S Web Service Annotation Framework. In *Proc. of the 13th Int. World Wide Web Conference (WWW 2004)*, New York, NY, USA, 2004.

[29] R. Pottinger and P. A. Bernstein. Creating a Mediated Schema Based on Initial Correspondences. *IEEE Data Engineering Bulletin*, 25(3):26–31, 2002.

[30] S. Staab, C. Tempich, and A. Wranik. REMINDIN': Semantic Query Routing in Peer-to-Peer Networks based on Social Metaphors. In *Proc. of the 13th Int. conference on World Wide Web (WWW 2004)*, New York, NY, USA, 2004.

[31] T. Strang and C. Linnhoff-Popien. A Context Modeling Survey. In *Proc. of the 1st Int. Workshop on Advanced Context Modelling, Reasoning and Management*, Nottingham, England, 2004.

[32] R. V. Guha and R. McCool. TAP: a Semantic Web Platform. *Computer Networks*, 42(5):557–577, 2003.

[33] S. Voulgaris, D. Gavidia, and M. van Steen. CYCLON: Inexpensive Membership Management for Unstructured P2P Overlays. *Journal of Network and Systems Management*, 13(2), 2005.

[34] D. Zeinalipour-Yazti, V. Kalogeraki, and D. Gunopulos. Exploiting Locality for Scalable Information Retrieval in Peer-to-Peer Networks. *Information Systems*, 30(4):277–298, 2005.

[35] L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, and H. Chang. QoS-Aware Middleware for Web Services Composition. *IEEE Transactions on Software Engineering*, 30(5):311–327, 2004.