A 2-tiers p2p Architecture to Navigate the Learning Objects Sea

Andrea Clematis, Paola Forcheri, Alfonso Quarati

Istituto di Matematica Applicata e Tecnologie Informatiche del CNR, Genova, Italy
Email : {Clematis, Forcheri, Quarati}@ge.imati.cnr.it

***Abstract* - Learning object repositories (LORs) are a strategic asset for e-learning. In a quite short period the scenario of use has quickly changed from an initial scarcity of learning objects (LOs) to an overabundance disorienting users. To address this problem, we present a conceptual model which leverages on the creation and sharing of comments about the use of LOs as experimented by learners. These comments originate a superstructure on interrelated and distributed LOs in the form of an annotated graph-of-comments. By relying on the graph structure, an ordering feature (based on an automatic ranking of LO) and a mechanism for extracting personalized views (subgraphs and paths in the graph-of-comments) are provided. Observing that both LORs and users are expected to belong to (possibly) different and autonomous communities, which share data via Internet, we propose a two-tiers architecture based on hybrid peer-to-peer technology efficiently realizing the navigation on LORs.**

***Index Terms* - Distributed Architectures, Collaborative and Personalized Learning, Learning Object Repositories**

I. INTRODUCTION

There is a growing interest towards the realisation of learning object repositories (LORs) aimed to facilitate the sharing of didactic material . LORs in fact constitute a kind of specialised digital libraries where high quality re-usable materials can be selected by users, on the basis of a description of their content (usually called metadata) as to educational features, context of use, technical aspects, and so on. This fact allows not only an efficient deployment of on-line courses but also it provides users with an effective means to share proposals and resources of learning aimed to improve traditional classroom activities, especially when repositories are enhanced with structures apt to allow the transmission of didactic ideas and experiences [2]. The development of the Web (Web2.0) and the availability of distributed systems represent an opportunity for the e-learning community [3], and the availability of learning objects on the web increases their significance and potential impact on different users.

Taking into account this fact, we focus attention on the problem of providing an adequate environment that permits to a distributed community of users to deal with distributed learning objects in an efficient and effective way. To this aim, we outline the interaction patterns that occur in a community of practice centred on LOs and enrich LOs with the expressive power deriving from such interactions. To support them, we presented in [4] a p2p based architecture that realises a network of distributed and inter-related user-defined comments apt to explore the content of LORs. These comments express observations and relationships between LOs as experimented by learners. This idea and the infrastructure designed have been then refined so to be used as a basis to address two further problems: 1) to rank LOs obtained as a result of a query, 2) to group LOs according to different types of user's objectives. Comments, in fact, originate a superstructure on interrelated and distributed LOs in the form of an annotated graph of comments. Vertex of the graph are the experimented LOs; annotated arcs connecting LOs explicit different relationships (e.g. similarity, specialization, complementarity) between distributed LOs as envisaged by users. By relying on the graph structure, an ordering feature (based on an automatic ranking of LO) and a mechanism for extracting personalized views (subgraphs and paths in the graph-of-comments) are provided. Ranking LOs resulting from a query is a quite relevant problem, as already noted by other authors [5]: queries in fact, especially in the case of multiple search in interrelated repositories, can produce a high number of results, thus asking users considerable labour and effort to find out relevant material of their interest. To address this problem, we observe that, in several cases, learners and teachers are used to speed up this task taking into account others' experience. Accordingly, we rank LOs on the basis of both the opinions and the associations with other LOs dynamically provided by the prior users. Ranking, however, is not enough to discover sets of objects that are relevant with respect to the task to be performed by a user who deals with a LOR. Repositories, in fact, can be fruitful source of resources for a number of learning-related activities: for example, users can look for ideas to design a lesson, or a group of lesson, in a traditional or blended context; some others are aimed to self-learning on the teaching of a well defined topic; or, they would like to find out suggestions for structuring and deepening the learning of a topic; in other situations, they intend to collect ideas for realising interdisciplinary approaches, and so on. In their current practice, users often ask their colleagues for hints and orientations to address these kinds of tasks. To provide a similar kind of support, we exploit again the network of user-defined comments, and rely on the experience of prior users to assist the user in finding out

doi:10.4304/jetwi.2.1.3-10

resources of interest. In our case, such experience is constituted by associations among LOs dynamically built by prior users and by the semantics they assign to these associations.

These ideas will be discussed in the following sections. More precisely: Section 2 presents our view of LOs and shows the ranking function and the kinds of personalised views that can be provided by means of this interpretation; Section 3 illustrates our architectural proposal along with a brief survey on p2p technology; Section 4 describes the use of such architecture for dealing with LOs' comments; finally, Section 5 concludes the paper with some remarks on the proposal.

II. Interacting with learning objects

To guide and support learners in the selection of the educational material most suitable for their need, we propose to establish an interconnection network linking LOs (possibly) belonging to different LORs. This network is characterized by the learning experiences developed by prior users through the use of LOs. To this end, we suggest that LOs should be enriched with the expressive power deriving from the interactions with both the users and other LOs.

By analyzing the prevalent kinds of interactions between a user and a given learning object lo, we introduce two levels of user–learning-object relationships:

A *reflection level,* represented by the variety of users' perceptions about *lo*. These perceptions may further refined as:

- User opinion (u), a non-qualified comment of a generic user;
- Peer review (p), the opinion of an expert officially entrusted with the task;
- Results of the experience (e), the description of a realm where *lo* has been used and the students' reaction;

An *interconnection level*, represented by the conceptual network of LOs, including *lo*, dynamically created by users during the search and the interaction processes. The kinds of relationships defined by users may be classified as:

- Specialisation (s), a learning object λo is indicated by a user as a specialisation of *lo* if, for example, the user thinks that λo could be used to go in deep or to show an example of a concept which is tackled by *lo*;
- Complementary (c), learning object λo is indicated by a user as a complement of *lo* if, for example, the user thinks the two LOs can be coupled in the same context or she/he experienced this use;
- Affinity (a), learning object λo is indicated by a user as similar to *lo* if the user thinks that λo and *lo* could be used indifferently.

This view leads us to interpret each LO as an annotated graph of both the connections between users' reflections on it and the interconnections, as seen by the users, between the LO at hand and other LOs in the network (its *neighbours*). LOs are identified by vertexes. Interactions between user and neighbours objects are identified by means of *interaction arcs*, labelled by tuples of the form *<User, Relation, Comment>,* where:

- *User* is any suitable reference to identify the user annotating the object;
- *Relation* is the identifier of the relation being established for the object, where *Relation* Є {*u,p,e,s,c,a*};
- *Comment* is the annotation associated to the relation.

The graph of Figure 1 depicts five relationships involving learning object LO_p and its two neighbours LO_q and LO_t. The three self-arcs outline two user opinions and a peer-review, created by users U_k, U_j and U_i respectively: a specialisation suggested by user U_i about learning object LO_q w.r.t. LO_p (denoted by a directed arc); and a complementarity with learning object LO_t individuated by user U_j (by a bi-directed arc). For the sake of simplicity the *Comment* elements are omitted in Figure 1.

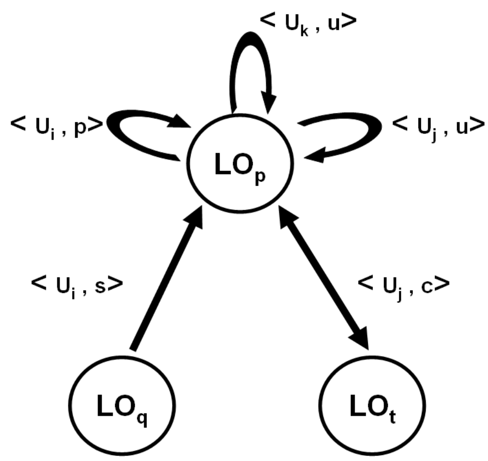

Figure 1 The annotated graph for learning object LOp.

By considering all the commented LOs, shared by the learning community, and their annotated graphs as a whole, we may trivially define the *graph-of-comments* as a network of interrelated user-commented LOs, that enforces, with new meta-knowledge, that provided by the distributed and autonomous LORs.

A. Searching for LOs

The *graph-of-comments* represents a conceptual model, that we superimpose over logically and physically separated structures (the connecting LORs), generally owned by third organisations and managed through tools (e. g. Learning management systems - LMS) that provide the creation and updating of LOs, only under precise utilization constraints and strictly access policies. Even the LOs discovery process is bounded by the modalities allowed by LOR managers (e.g. metadata based query). For these reasons, it is no feasible to assume any architectural hypothesis or any other constraint over the typology of LORs. The definition and implementation of our network is therefore orthogonal to LOR. Thank to this network, as we show in the following, LOs are to be queried independently from the discovery mechanisms offered by the underlying LMSs. Relying on the *graph-*

of-comments, LORs may be queried through the bare-content of the comments, via some reference about the author of an annotation (e.g. "find all LOs commented by Prof. E. Smith") or by using the interconnection relationships among LOs envisaged by prior users (e.g. "find all material similar to the LO at hand"). The neat separation and the reciprocal independence between LORs and the *graph-of-comments*, allow the definition and realization of a portable and autonomous infrastructure. To facilitate users in exploring the graph-of-comments, thus sailing the LOs sea, we propose two sets of instruments (integrating the core discovery tools): comments-based ranking of LOs and personalized roadmaps.

B. *Ranking LOs*

The use of ranking to help the user to select suitable learning objects is proposed in [5], with the creation of a ranking function to access the relevance of learning objects according to a specific user and context. Merlot [6] and DLNET (http://www.dlnet.vt.edu) provides tools to rated LOs on a five-point scale. The majority of these rating mechanisms require LOs users to explicit express their vote on LOs. By contrast, our solution bases his ranking facility in automatic, by examining the overall comments about LOs along with their relationships and obtaining several corresponding metrics.

The process is as follows: as result of the comment activities, each LO is associated to a set of weights (one for each of the six relationship involving LOs), namely the *ranks vector*, accounting for the number of comments it received by users. Similarly to PageRank™ used by Google to evaluate search-results on the Web, these weights measure the degree of popularity of a fixed LO among the learning communities, and supplies users with a useful quantitative index. Ranks along with comments (qualitative index) establish a new, and more exhaustive user-centric view of LOs and of their dynamic interrelations. Ranks vectors are used to order the results of a user's query. Note that, if desired, it is possible to order respect each of the six vector components.

For learning object LO_p of Fig. 1, we have the ranks vector depicted in Table1, where the last column (rank) is the number of all comments expressed on LO_p.by users.

Table 1 Popularity measures for learning object LO_p: each column, but last, corresponds to one of the six user-lo relationships

	u-rank	**p-rank**	**e-rank**	**s-rank**	**c-rank**	**a-rank**	**rank**
LO_p	2	1	-	1	1	-	5

Differently from other systems, the rank for a given LO is not given by averaging the votes from distinct users to obtain a relative value, but by assigning an absolute "popularity" weight. It may be objected that this ranking technique does not capture the "real" willingness of the user to suggest or discourage the use of a LO, and therefore it could happen that a LO strongly criticized may rank better than a recommended one. However we argue that the relative interest of a LO with respect to another may result in its absolute rank, observing that an heavily hit LO is probably more pedagogically interesting. More appropriately, it is reasonable to think that people is willing to express their comments (wasting their time on this activity) on valuable items. It is arguable whether an explicit expression of a vote (in a graduate scale) is always more objective and less prone to mistakes or misinterpretations. In fact, differently from an hotel, a wine, a web-mall, where an exact judgement is more immediate and under certain limits quite unquestionable (bad, good, excellent), an experience with a LO may regards multiple perceived aspects of the learner involved. In any case the practice of use of the system can supply precious insights on this aspect, and we don't exclude to introduce, in the future, some form of explicit vote for the cases of peer reviews and user opinion.

C. *Personalised roadmaps*

The per-object view of interconnections among LOs can be combined in three different graphs (namely *interaction graphs*) that summarize at the community level the network of relationships among objects. The *specialization* graph is a direct graph that allows to identify objects that constitute a specialization process from some general content to a set of specific information. Thus, it allows the user to follow a learning path in a knowledge domain. The *complementary* and *affinity* relations generate undirected graphs, since complementary and affinity interconnections identify symmetric relations between two objects. By means of the complementary relation it is possible to get an help in order to identify a set of LOs that represent available resources on a given domain. One of the possible uses of the *affinity* graph is to set up a catalogue of learning objects that have a similar content and approach in addressing a selected topic.

By visiting an *interaction graph* the user may avail herself of the experience of previous 'navigators'. The main idea is to supply learners with some sort of personalised roadmap, to help her to find the 'right way', starting by the node at hand, namely the *root*, to reach new and possibly unexpected destination, without having to blind-moving in the LOs web. The meaning of *personalised* is twofold: it refers to the individual and actual perspective of the exploring user who chooses her own direction, as well as to the previous personal observations of others who traced the path. To aid the learner to choose between alternate paths in a graph, the graph view may further be enriched by weighting each node by the ranks.

III. A 2-TIERS P2P ARCHITECTURE TO ENABLE THE SPREADING OF LOS

A. *Hybrid peer-to-peer architectures*

The *graph-of-comments* establishes a platform for distributed learning, allowing users hare their experiences and possibly cooperate. Moreover, the mass of comments about LOs, generated by independent stakeholders, asks for strong requirements on management and organization of content. These aspects lead us to consider peer-to-peer architectures as natural candidates to represent the supporting environment for distributed comments on learning objects, preferable in our opinion to centralized solutions, that suffer of several drawbacks (e.g. bottleneck, scalability, single point of failure).

Peer-to-peer emerged as a light-weight interaction paradigm finalized to share resources among individuals in a rapidly changing environment. Thanks to its collaborative paradigm, peer-to-peer attracted the attention of researchers involved in e-learning and different proposals [7] and research projects, such as Edutella [8], where developed. Various P2P paradigms arose in the last few years. A common way to distinguish among them is to examine the nature of the overlay (structure) they superimpose on the underlying network of participating peers. Considering their structural design, P2P networks range from unstructured ones to structured networks, through intermediate or hybrid P2P systems using various mixed approaches.

Unstructured or *pure* p2p networks, such as Freenet (http://freenet.sourceforge.net) or Gnutella (http://www.gnutella.com), are truly distributed systems in which all the nodes are equal and there does not exist any form of centralization. In principle, these systems are scalable and fault tolerant, and give good responses to keyword searches (i.e. given a sequence of keywords, find files that match them). However, caused by their completely random structure, they turn out to have some heavy performance limitations, e.g. due to the necessity of interacting with all the nodes of the network when executing a query (network flooding). Another weak point is that, being the environment heterogeneous, slow nodes may become the bottleneck of the system.

Structured P2P, such as Chord [9] or CAN [10], are based on Distributed Hash Table (DHT). In this case both peer-nodes and searchable resources are mapped through a hash code over a structured key space (e.g. a ring, a tree). While DHTs are scalable, self organizing and efficient solution, which straightforwardly allow to perform exact-match lookup operations of resources using a single attribute; they are les adept at supporting range queries, multi-attribute queries and more general keyword searches. Moreover structured topologies may require high maintenance cost especially in the presence of a high churn rate.

Early examples of *hybrid* peer-to-peer architectures introduce some form of centralization at least to keep an index or directory of relevant information. Super Peer Networks (SPN) [11] present the concept of super-peers, i.e., peers with additional capabilities, is introduced in these systems. Super-peers (SP) nodes form the backbone of the overlay network having normal peers placed around them. SP nodes act as a query server with respect to a set of clients (normal peers) and as peers with respect to the other super peer nodes. Each client node is connected exactly to one super peer node. This means that each client to post a query has to interacts only with its super-peer node, which in turn interacts with other super peers in a canonical Peer-to-peer fashion (i.e. by passing requests to its neighbors SPs).

Other hybrid solutions aimed to achieve the results of structured and unstructured p2p networks by combining their technical and topological solutions have been proposed: Brocade [12] is a hybrid overlay network proposal, where a secondary overlay is layered on top of a primary DHT; SHARK (Symmetric Hierarchy Adaption for Routing of Keywords) [13] employs a hybrid DHT solution for rich keyword searching.

B. *The proposed architecture*

Being informed by collaborative-pedagogical requirements (e.g. sharing, mining, and updating heavyweight comments widespread on the network) and by technological constraints (e.g. distributed nature of the learners, heterogeneity of clients machines) emerging from the community of LOs users; our choice to explore the *graph-of-comments* in an efficient and effective way, led us to the design of a *two-tiers* p2p architecture, following an hybrid approach. The first overlay (namely Tier 1) provides the ground to develop the basic mechanisms for sharing and discovery comments about LOs; the second overlay (Tier 2) supplies the necessary infrastructure to the distribute storage of the graph-of-comments and the ranks-vector associated to LOs.

More in detail, Tier 1 relies on a super-peer node infrastructure. User-created comments on LOs are managed by simple peer nodes which store them. Besides, the information labelling the interaction arcs relating to the commenting LO is transmitted to the SP node responsible for the publication on the network and interacting with other SP nodes in the discovery process. The connection between SP nodes of Tier 1 is unstructured, so allowing the execution of multi-attribute and range queries, in a straightforward way respect to that offered by DHTs topologies.

The structured network realized on the second overlay (Tier 2) offers the support to order the set of results (based on ranks) and to extract personalised paths from the graph-of-comments. Tier 2 establishes a DHT infrastructure upon the super-peer nodes of Tier 1. Figure 2 shows the different neighbouring relationships established by the two tiers on the SP nodes (e.g. node A distances 1 hop from D in Tier 1 and 3 hops in Tier 2). Normal peers are attached to each SP node of Tier 1 and SP nodes are link each other without any ‘particular’ topology. In Tier 2 SP nodes a connected through a ring.

To build up the graph of comment, the URI of each commented LOs is hashed and indexed in the appropriate node of the distribute table along with its rank vector and all its neighbours. Thanks to this layer, the ranks vector of a LO is updated as soon as a new comment on the LO is created by a user.

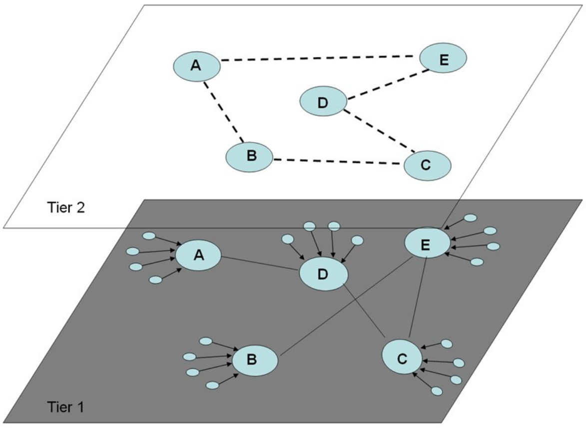

Figure 2 Tier 2 overlay, organized as a DHT, superimposes on the unstructured SP network of Tier 1

As known, a DHT requires O(log N) hops (with N number of nodes in the network) to look-up a certain key. We are able to extracts or update the ranks vector or the neighbours of a given LO in O(log *spn*), where *spn* is the number of SP nodes. Based on this look-up primitives and adopting graph-visit algorithms (e.g. Breadth first), it is possible to construct for each given LO one of the three interconnection graphs rooted on it.

It is important to say that SPN as a support architecture to share LOs has been early considered by researchers involved in the Edutella project [14]. However while their emphasis is mainly focused on the efficiency of the query mechanism, by superimposing to the SPN a DHT overlay structure we are able to supply users with mechanisms for an ordered selection of LOs and with a feature to build-up personalised roadmaps based on prior users experiences.

IV. Working with comments about LOs

To navigate the *graph-of-comments* we consider two *canonical* classes of queries. The first kind mines the content of comments to obtain some useful LOs: for example the query "find all LOs related to comments speaking about the Napoleon's campaign in Italy", will return the ordered list of LOs associated to relevant comments. These queries may integrate or even substitute the search tools offered by the LORs, and the content of a comment acts like an 'alias' for the associated LO. The second class of queries is of the form: "find comments about *topic*". This second case is more immediate respect the first one: comments are queried per-se, and it is probable that users are more interested in the opinion of other learners about a learning experience than in the related LO itself (that it is always possible to return, to complete the result set).

Indeed, a third typology of query can be envisaged: the ones aimed to obtain a personalised roadmap, through one of the three interaction graphs (see II.C), starting from a given *root* LO. This case is more specific and essentially concerns three fixed queries reflecting the pattern: "return all the LOs *relationship root*" where *relationship* is one of 'specialising', 'complementing' or 'similar to'. More important from an architectural point of view, these 'relationship' queries only involve Tier 2 as they uniquely depend on the relationship between LOs established by users and therefore do not require any hypothesis on the format of the comments or of the data-structures used to store them, assumptions that are necessary to execute queries of the first two classes.

The kind and the granularity of the information that need to be spread out in the two-tiers framework is greatly influenced by the extent of the mining capability of the network of LOs. Depending on the solution adopted, it will be possible to query LORs in a more sophisticated way or not. If a full text search is desired, inverted index technology [15] should be adopted: for each term contained in a comment, a list of the identifiers of comments that contain that term has to be pre-computed and stored. If a query involves more terms the related lists are intersected. In all other cases where full text format is not feasible (e.g. comments in audio or audiovisual format) or required, the discovery of comments may be driven by metadata about the content of comments [8].

Starting from these observations, let us consider some scenarios of use of the 2-tiers infrastructure proposed. To exemplify the system behaviour in response of users' activity, we introduce a reduce set of services. Table 2 lists the services involved for each of the three main tasks, also specifying the responsibility for each type of network node.

Table 2 Main services to buidld-up and explore the 2-tiers infrastructure

	Create a comment	**Canonical query**	**Personalised roadmap**
Peer	`CreateComment`	`DiscoveryAgent`	`BuildRoadmap`
SP node Tier 1	`T1Indexer`	`T1Discovery` • `T1LocalSearch` • `T1Query Dispatcher`	
SP node Tier 2	`T2UpdateRank&BuidlRelation`	`T2DHTRankLookUp`	`T2BuildPaths`

A. *Creating comments*

To create a new comment the `CreateComment` service of the peer is invoked by a user. After the editing, the entire comment is saved on the user file system, and the information labelling the interaction arc, namely the *comment-tuple*, is built. For a comment tuple $c = <u,r,src,dst,fid,Data>$, we have: r identifies one of the six relationships of Section II ($r \in \{u,p,e,s,c,a\}$); *src* and *dst* are the URIs of the learning objects (source and destination) involved in the relationship (the same URI if r is one of the three "Reflection" relationships); u identifies the commenting user; and *fid* denotes the identifier of the file storing the comment inside the local peer file system. The *Data* field is the extra info required to execute searches on the content. As explained above, *Data* may range from a set of specific metadata to the complete inverted list of the words belonging to the comment, needed to exploit a full text search. In either cases *c* is advertised to the network, by calling the `T1Indexer` service that resides on the SP node responsible for the peer. `T1Indexer` is in charge of accomplishing all the indexing operations necessary to respond to canonical queries (e.g. create and maintain the overall inverted lists of all the comments indexed by the peers managed by the node), and to build up the Tier 1 overlay, by organizing and storing list of adjacent SP neighbours.

Moreover, Tier2 is updated by invoking `T2UpdateRank&BuidlRelation` on the same SP node. Given the LO *src,* for which the relation *r* is being established, the key *k*=*hash*(*src*) is computed (where *hash* is the hashing function adopted by the DHT environment). Let spn_k be the SP node in the DHT responsible for the key *k* (we remind that in a DHT the nodes are mapped to the keys space). If *src* is in spn_k (the *src* vertex has already been inserted in the graph-of-comments), the ranks vector of *src* is update by adding 1 to the *r* component. Otherwise, a new vertex with label *src* is created and the *r* component of the associated ranks vector is set to 1. Moreover, if $r \in \{s,c,a\}$ (that is, $src \neq dst$) a new arc from *src* to *dst* is inserted from each of the three transaction graphs. If *r* is symmetric (that is $r \in \{c,a\}$) one arc from *dst* to *src* is also inserted in the node responsible for *hash*(*dst*)). The total cost of this operation, in number of exchanged messages, is O(log(*spn*)) (related to the lookup in the DHT of nodes *src* and *dst*).

B. Looking for comments and LOs

Depending on the granularity of the *Data* field contained in a comment-tuple (that also determinates the associated data-structures and supporting algorithms) it will be possible for users to express canonical queries according to a keyword (full-text) or a metadata approach; in either cases, the core query-mechanism is the same. A query is submitted by users by invoking the `DiscoveryAgent` service running on users' peers. To reduce latency, occurring when visiting the whole network, we allow users to set a threshold (*thres*) to the total amount of objects returned. The `DiscoveryAgent` is a two-phases service that, at first, looks for satisfying objects by exploring Tier 1; then, it orders the results relying on the rank vectors managed by Tier 2. The first phase is initiated by calling the `T1Discovery` service residing on *S*, the SP node responsible for the query-emitting peer. This service (generally) executes a local and a global search. The first look-up is executed by `T1LocalSearch`, that checks whether the required objects (being comments or LOs references) are identified in the comment-tuples stored and indexed in *S*. If *rs* objects are found and *rs* > *thres;* the search on Tier 1 is finished and `T1Discovery` returns. Otherwise, a call to `T1QueryDispatcher`, aimed to retrieve the remaining *thres*-*rs* objects, is activated. `T1QueryDispatcher` is based on a random-walk algorithm [16] that, starting from some of the neighbours of *S*, visits the network. For each of the neighbours visited, a call to the residing `T1Discovery` service is made. The discovery process terminates when *thres* objects are found or the network is entirely examined. Once the `DiscoveryAgent` gets the result set, it invokes `T2DHTRankLookUp` which is in charge of the rank-based ordering of the resulting comments. To this end, for each comment-tuple, in the result set, the referred LO is extracted and its ranks vector is retrieved from the DHT ring of Tier 2.

C. *Building personalised roadmaps*

To construct a personalised roadmaps rooted at a given LO *lo*, the user invokes the `BuildGraphs`(*lo*,*r*) service, with *r* identifying the desired interaction graph. `BuildGraphs`, running on the user's peer, calls the `T2BuildPaths` service residing on the corresponding SP node, which starts the construction algorithm (namely a Breadth first search) by visiting the graph-of-comments stored in Tier 2 and returning a spanning tree, rooted at *lo*, to the requiring peer.

D. *Early* ***evaluation***

The above discussed 2-tiers structure and in particular the look-up functionalities may be validated mainly by considering the overall number of LOs, the quantity and the granularity of the data managed by each super-peer node and the total amount of super-peer nodes composing the first overlay (Tier 1). Performance are also affected by threshold values limiting the number of objects returned and hence the number of possible network hops. Moreover as already pointed-out considering that a DHT requires O(log N) hops (with N number of nodes in the network) to look-up a certain key, it is possible to extracts the ranks vector or the neighbours of a given LO in O(log *spn*).

CONCLUSIONS

Learning Objects are an important asset in every e-learning community. Their availability in a distributed Web-based environment represents an opportunity for further developments generated by a richer context of interaction that permits to different stakeholders to use, comment and evolve available objects, thus providing each other with suggestions and insights.

In this paper our attention is to provide an adequate environment that permits to a distributed community of users to interact with distributed learning objects in an efficient and effective way.

To help LORs users to overcome the feeling of disorientation in dealing with a high mass of LOs, our proposal leverages on users-created comments about LOs, that enrich them with mutual relationships envisaged by users.

On this basis we supply users with mechanisms for an ordered selection of LOs and with a tool to build-up personalised roadmaps based on prior users experiences.

Observing that both LORs and users are expected to belong to (possibly) different and autonomous communities, which share data via Internet, we propose a two-tiers architecture based on hybrid peer-to-peer technology efficiently realizing the navigation on LORs.

At present we acknowledge the need to carry on a deep analysis of the behavior of our framework in real scenarios and we planed an experimentation on the field as next step of our work.

References

[1] Margaryan A., Littlejohn A., "Repositories and communities at cross-purposes: issues in sharing and reuse of digital learning resources", Journal of Computer Assisted Learning, 24, 2008, 333–347.

[2] Busetti E., Dettori G., Forcheri P., Ierardi M.G. (2008), "From LOs to educational itineraries: helping teachers to exploit repositories , Advances in Web Based Learning" - 6th International Conference, Revised Papers, Leung, H., Li, F., Lau, R., Li, Q. (Eds.), Lecture Notes in Computer Science, Vol. 4823, Springer, 2008, 496-507.

[3] Li Qing, Lau R.H., Shih T. K., LI F. W. B., "Technology supports for distributed and collaborative learning over the internet", ACM Transactions on Internet Technology, 8 (2), 2008, 1- 24.

[4] Clematis A., Forcheri P., Quarati A., "A p2p framework for interacting with learning objects", in Creating New Learning Experiences on a Global Scale, Duval, E., Klamma R., Wolpers M. (Eds.), Lecture Notes in Computer Science 4753, Springer, 2007, 469-474.

[5] Ochoa X. , Duval E., "Relevance Ranking Metrics for Learning Objects", IEEE Transactions on Learning Technologies, 1 (1), 2008, 34 – 48.

[6] Nesbit J., Belfer K., Vargo J., "A Convergent Participation Model for Evaluation of Learning Objects", Canadian Journal of Learning and Technology, 28(3), 2002, 105-120.

[7] Yang, S. J. H., "Context Aware Ubiquitous Learning Environments for Peer-to-Peer Collaborative Learning", Educational Technology & Society, 9 (1), 2006,188-201.

[8] W. Nejdl, B. Wolf, C. Qu, S. Decker, M. Sintek, A. Naeve, M. Nilsson, M. Palmer, T. Risch, "Edutella: A P2P Networking Infrastructure Based on RDF", Proceedings of the 11th international conference on World Wide Web, 2002, pp. 604-615.

[9] I. Stoica, R. Morris, D. Karger, M. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications", ACM SIGCOMM 2001, 2001, 149-160.

[10] Sylvia Ratnasamy, Paul Francis, Mark Handley, Richard Karp, and Scott Shenker, "A scalable content-addressable network", ACM SIGCOMM 2001, 2001, 161-172.

[11] Yang, B., Garcia-Molina, H., "Designing a super-peer network", In: Proceedings of the 19th International Conference on Data Engineering., 2003, 49–60.

[12] Ben Y. Zhao, Yitao Duan, Ling Huang, Anthony D. Joseph, John Kubiatowicz, "Brocade: Landmark Routing on Overlay Networks", IPTPS 2002, 2002, 34-44.

[13] Mischke, Jan, Burkhard Stiller, "Rich and Scalable Peer-to-Peer Search with SHARK", Active Middleware Services, , 2003, 112–121.

[14] W. Neidl, M. Wolpers, W. Siberski, C. Schmitz, M. Schlosser, I. Brunkhorst, A. Loser, "Super-peer-based routing strategies for RDF-based peer-to-peer networks", Journal of Web Semantics, 2004, pp.177-186.

[15] Justin Zobel, Alistair Moffat, "Inverted files for text search engines", ACM Comput. Surv. 38(2), article number 6, 2006, 56 pages.

[16] Qin Lv , Pei Cao , Edith Cohen , Kai Li , Scott Shenker, "Search and replication in unstructured peer-to-peer networks", Proceedings of the 16th international conference on Supercomputing, 2002, 84-95.

Andrea Clematis is a Research Director at the Institute of Applied Mathematics and Information Technologies of Italian CNR in Genova (IMATI-CNR). His research interests are in the area of high performance, parallel and distributed computing, with emphasis on programming environments and applications of parallel computing. He co-authored more than 120 technical papers, published in journals, book chapters and conference proceedings. Dr. Clematis acted as project leader for IMATI in about 20 cooperative research projects founded by different institutions including European Commission, Nato, and British Council. He is the Chair of the Euromicro Technical Committee on Parallel and Distributed Processing, and the subject area editor for Parallel and Distributed Processing of the Journal of Systems Architecture (Elsevier).

Paola Forcheri is Research Director at IMATI-CNR, where she is working since 1976. Her research interests mainly focus on technology enhanced learning, with particular reference to collaborative learning models and approaches based on multimedia and networking technology, and pedagogical metadata models. Application fields of the work are mainly teacher education and continuous training. On these and related themes Paola Forcheri has been involved in numerous international and national project, also with a role of leader. She has authored numerous papers at national and international level, registered several trade-marks, edited special issues of journals and conference proceedings, take part in the scientific organisation of international and national events, is included in the editorial and reviewers boards of international journals. (http://www.ge.imati.cnr.it)

Alfonso Quarati is Researcher at the Institute for Applied Mathematics and Information Technology (IMATI) of the Italian National Research Council (CNR) in Genoa since 1997. His current research interests cover models and tools for grid computing, peer-to-peer computing, collaborative working technologies, e-learning. He co-authored more than 40 scientific papers, published in journals, book chapters, technical reports and conference proceedings