# CLOSER: A Collaborative Locality-aware Overlay SERvice

Marco Papa Manzillo, Luigi Ciminiera, Guido Marchetto, Fulvio Risso, *Members, IEEE*

**Abstract**—Current Peer-to-Peer (P2P) file sharing systems make use of a considerable percentage of Internet Service Providers (ISPs) bandwidth. This paper presents the Collaborative Locality-aware Overlay SERvice (*CLOSER*), an architecture that aims at lessening the usage of expensive international links by exploiting traffic locality (i.e., a resource is downloaded from the inside of the ISP whenever possible). The paper proves the effectiveness of CLOSER by analysis and simulation, also comparing this architecture with existing solutions for traffic locality in P2P systems. While savings on international links can be attractive for ISPs, it is necessary to offer some features that can be of interest for users to favor a wide adoption of the application. For this reason, CLOSER also introduces a privacy module that may arouse the users' interest and encourage them to switch to the new architecture.

**Index Terms**—P2P, file-sharing, traffic locality, privacy

## 1 Introduction

**Motivation:** Peer-to-Peer (P2P) file sharing systems have been experiencing a constantly increasing popularity during the last decade. This success is driving an evolution of these systems in terms of scalability, reliability, and decentralization. From Internet Service Providers (ISPs) point of view, file-sharing systems are both an opportunity and an issue: while these systems are a major driver for high-speed residential subscriptions, they force ISPs to increase their infrastructure bandwidth very often and, first of all, purchase more expensive transit services from Tier 1 carriers.

A promising approach to solve this problem consists in modifying one or more system components (e.g., the user application or the indexing system) in order to attempt directing requests to the closest peers that own the requested resource (referred to as *resource providers* in the following). Examples of solutions adopting this method are presented in [1]–[5]. However, these solutions are suboptimal from a traffic locality perspective as in the selection of possible resource providers to contact for download they can consider only a subset of the available peers, thus potentially excluding some local providers. This is due to some design choices made in these solutions, which for scalability reasons cannot have access to the localization information of all the available resource providers (see Section 3.1 for details). Moreover, these systems do not give an adequate importance to the central role that users have in the evolutional process of P2P systems. In fact, a significant percentage of the most widely used P2P applications has been developed and maintained by user communities, which need to be motivated to collaborate at the dissemination of novel systems and paradigms. At first sight, the locality-awareness seems to offer an intrinsic benefit for users that could stimulate their cooperation: since it reduces the average number of network hops crossed by download connections, it is statistically harder to traverse a bottleneck link and, consequently, the average download time should decrease. However, several publications [1], [6]–[8] demonstrate that this is not true in general and that in certain situations the download time may rather increase. Hence, we need different incentives which, similarly to the download time, are of interest for users. Without these incentives, the locality-aware techniques carry uneven advantages for ISPs and users, which may drastically limit their adoption in the existing P2P communities.

**Contributions:** Basing on these considerations, we developed CLOSER (Collaborative Locality-aware Overlay SERvice). CLOSER improves existing locality-aware solutions by offering the guarantee for downloads to be executed locally whenever is possible — i.e., when the resource is present in the requester neighborhood — as it can discriminate among all possible resource providers when operating locality-aware selections. This is obtained with a negligible effort for the ISP and without affecting the scalability of the locality-awareness approach. The proposed solution is evaluated by both analysis and simulation, which also demonstrate the real importance of using the complete list of resource providers in the locality-awareness context. Furthermore, the feasibility and the simplicity of the approach are verified through the development of a real CLOSER-aware P2P application. CLOSER also introduces a novel mechanism to anonymize users' behavior in the network in order to stimulate their cooperation and hence favor the spread of the solution. This choice is motivated by the significant number of attempts to build *anonymous P2P systems* (e.g., [9]–[12]) driven by open source, users supported, communities, which can be an indicator of users' vivid interest for privacy. These solutions are based on the utilization of proxy nodes as intermediaries during re-

*The authors are with the Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy (email: name.surname@polito.it).*

source downloads, which guarantee anonymity, but to the detriment of the download speed [13]. The proposed privacy module overcomes these limitations by enabling direct downloads and it is shown not to violate the locality-awareness principle.

**Outline:** The paper is organized as follows. Section 2 presents the most prominent solutions applying the locality-awareness principles. The CLOSER architecture is described in Section 3, while Section 4 presents the privacy module that CLOSER includes to encourage users to change their P2P applications. Section 5 provides some analytical results concerning the effectiveness of CLOSER in lowering the inter-ISP link utilization, while Section 6 illustrates the simulation scenario and reports some simulation results showing the benefits stemming from the proposed solution. Finally, Section 7 concludes the paper.

## 2 Related work

Several possible solutions exist to provide traffic locality in P2P file-sharing systems. A promising approach consists in directing requests to the closest resource providers through the modification of some components of P2P systems, possibly in conjunction with the deployment of additional modules that slightly modify current P2P paradigms. Since also CLOSER belongs to such category, solutions adopting this method are briefly described in the following. Different approaches and their main drawbacks are instead presented in Appendix A, which can be found in the Supplementary File.

A first approach consists in modifying the behavior of current P2P applications, so that they can autonomously acquire their localization information and provide it to other users interested in the resources they share. In essence, a node acquires the list of resource providers from the indexing system. Then, it contacts the resource providers present in the list asking them for their localization information and compares the obtained results with its own localization data. Closer resource providers are preferred to the distant ones. Examples of systems belonging to such category are Ono [1], a software extension of the Azureus BitTorrent client, and Kontiki [2], proposed in the context of P2P streaming. Each Ono instance determines its location by querying a Content Delivery Network (CDN) for a fake resource and collecting the mirror sites that the CDN chooses for it, according to the principle that users are redirected to a set of mirrors that are probably close to them (e.g., users always redirected to US mirrors are probably located in US). Kontiki implements a simpler localization methodology: starting from their IP addresses, Kontiki nodes obtain their AS Number (ASN) — assigned to ISPs by the Internet Assigned Numbers Authority (IANA) — from public databases.

A second approach consists in creating and exploiting a strong collaboration between users and ISPs, to be used in conjunction with some modifications to either the P2P application or the indexing system. In particular, each ISP deploys a special equipment providing the localization information to either the applications or the indexing system, depending on the specific solution. For example, [3] proposes to deploy a centralized equipment called *oracle* that users can query once they have acquired the list of resource providers from the indexing system. On the contrary, the P4P solution [4] proposes to deploy an *iTracker*, which is equivalent to the oracle facility but is directly contacted by the indexing system before sending the list of resource providers to a querying user. Thanks to the ISP collaboration, these techniques offer more precise localization information, with consequent improved performance in circumscribing traffic with respect to Ono and Kontiki. However, this results in an additional effort for the ISP, which has to deploy and maintain the equipment (oracle/iTracker) for ordering the list of possible service providers. It is also worth noticing how the presence of such equipment may allow malicious users to reconstruct the ISP topology — that generally is a confidential information — by forging ad hoc requests and analyzing the oracle/iTracker answers. On the other side, Ono and Kontiki do not require the ISP intervention and are less sensible to malicious peers aiming at reconstructing the ISP topology, but to the detriment of the localization precision.

More recently, these solutions have been used as a basis for additional work aiming at studying different aspects of the traffic locality problem. Considering a BitTorrent swarm, [14] and [15] investigated the impact of introducing the locality-aware principle not only in the neighbor selection, but also in the peer and piece selection procedures (i.e., the two operations which drive resource downloads in BitTorrent). Furthermore, [16] studied the adoption of BGP routing information as localization data used for ranking resource providers, while [17] explored the effects on the network when only a subset of resource lookups can be locality-aware. Also the IETF expressed interest in the topic by forming an Application Layer Traffic Optimization (ALTO) working group [5] for standardizing a protocol for traffic locality in P2P systems. The IETF solution is based on an ALTO server which can be contacted to acquire the locality information, thus following the oracle/P4P approach.

## 3 CLOSER

### 3.1 Rationale

All solutions described in the previous section have a common operating principle: the locality information related to the resource providers is acquired (either by the applications or by the indexing system) whenever a user starts a lookup for a given resource, i.e., at *lookup time*. In Ono, Kontiki, and the oracle-based solution, users acquire a list of resource providers from the indexing system, and then collect locality information related to the listed providers. However, for scalability reasons, indexing systems generally do not supply nodes with

an exhaustive list of resource providers, but randomly selects a subset of $L$ resource providers among all the available ones — by default, $L = 50$ in BitTorrent. Let us denote this list as *sampled list*. Since every ISP includes a small percentage of the Internet population, it is unlikely that the sampled list includes a high number of resource providers located in the same ISP. Hence, the optimization process executed by these techniques may not be very effective. A similar problem is present in P4P, as the indexing system can send only a "sampled list" of the available resource providers to the iTracker, which hence performs a suboptimal ranking. Let us denote this issue as *sampled list problem*.

The rest of this section presents CLOSER, which avoids the sampled list issue as it ensures to consider all possible resource providers when discriminating among them. Furthermore, as it will be clearer in the following, CLOSER ensures the localization information adopted to be precise with a slight overhead for the ISP, which does not have to maintain any specific infrastructure.

### 3.2 CLOSER overview

CLOSER locality-awareness relies on two main principles: (*i*) the indexing system is made aware of the localization information of every resource provider, and (*ii*) this is done by enabling resource providers to communicate their localization information to the indexing system whenever they register a new resource, i.e., at *registration time*. During a lookup procedure, a requester gives its own localization data to the indexing system, which, thanks to these operating principles, can directly sort the resource provider list by increasing distance from the requester. In this way, even if the indexing system has to limit the resource list sent back to the requester to $L$ entries for scalability reasons, the first $L$ entries are the most interesting from the locality-awareness point of view. Hence, if the indexing system can guarantee to locate all the available resource providers (e.g., a BitTorrent tracker or a DHT), it is possible to guarantee that if even a single resource provider is present within a given topological distance from the requester (e.g., in the same ISP or in the same country, depending on the adopted localization information), it will be sent to the requester with the correct associated distance. In essence, CLOSER enables a P2P system to discriminate among all possible resource providers without transferring the complete list. This is not possible with other solutions, which cannot use the complete list of providers for locality-awareness purposes as they should transfer the entire list over the network (generating the abovementioned scalability issues). Instead, our choice to move the localization data to the indexing system and their acquisition at registration time allows the locality-aware system to use the complete list in a simple and scalable way.

To make a P2P system CLOSER-aware, we need to modify both the indexing system — which has to be enabled to understand the localization information and sort the available resource providers according to this parameter — and the P2P application — which has to be able to interact with this new indexing system, referred in the following as *CLOSER indexing system*. Table 1 summarizes the modifications needed by current P2P systems to be compliant with the analyzed locality-aware techniques, including CLOSER. It is worth noticing that, although two separated columns are shown in the table for the modifications required by the P2P application and the indexing system, these two components coincide when the indexing system is distributed (e.g., a DHT such as Kademlia [18] or in Gnutella [19]), as the indexing system is built and maintained by the application itself. This is the case for the majority of modern P2P systems (including BitTorrent), which tend to migrate to decentralized approaches. In CLOSER, the localization information has to be stored at the indexing system together with the resource itself. However, the small size of this information (a few bytes are sufficient to represent these data) makes this to result in a negligible cost if we consider the storage capabilities of modern computer architectures.

TABLE 1
Summary of modifications needed by locality-aware systems

| System | ISP Support | Modified P2P Application | Modified Indexing System |
|---|---|---|---|
| oracle | Required | Required | No |
| P4P | Required | No | Required |
| Ono | No | Required | No |
| Kontiki | No | Required | No |
| CLOSER | Optional | Required | Required |

Additional details on CLOSER are provided in Appendix B and Appendix C, which can be found in the Supplementary File. In particular, Appendix B shows the resource registration and retrieve procedures, while Appendix C describes the structure for the localization information we thought for CLOSER.

### 3.3 ISP support

In CLOSER, the resource providers themselves communicate their localization information to the indexing system during the registration procedure of new resources.

Similarly to what has been proposed for Kontiki [2] (see Section 2), resource providers may acquire their localization data autonomously (e.g., by querying public databases such as GeoIP [20]), without any intervention from the ISP[1]. Although compliant with the CLOSER operating principles, this approach may reduce the accuracy of the localization information.

On the other hand, a proper ISP support can improve the system performance. In fact, if the ISP provides

1. These databases are useful for the higher level of the hierarchical localization information adopted in CLOSER. Lower hierarchical levels — e.g., the country and the town where the node is located — can be provided directly by users when starting the P2P application, since they usually know where they are located.

nodes with their localization information, this will result more accurate, thus allowing CLOSER to better achieve traffic locality.

To support CLOSER, ISPs do not have to deploy any infrastructure: they simply have to provide nodes with their localization data, which can be easily distributed through widely used systems — e.g., a web application. This is an advantage with respect to other techniques such as the oracle-based or P4P, which instead have to maintain specific servers. Furthermore, in CLOSER the ISP provides the localization information to each single resource provider and, thus, a malicious user should acquire localization data from every single user to reconstruct the ISP topology. This may be complicated as users' applications are not programmed to reply to direct queries concerning their topological information. This is another important difference with respect to the oracle/P4P scenario, where a malicious user can easily reconstruct the entire topology by simply interacting with the oracle/iTracker.

It is also interesting to remark that ISPs solely provide localization data; this produces some benefits: (*i*) future changes to P2P protocols do not require ISPs support and can be decided by the P2P application developers autonomously, solving the concerns highlighted by the P2P user community in [21]; (*ii*) users do not disclose information to ISPs or third parties, which the P2P user community highlighted as an issue in [22]; (*iii*) there is no legal concern for ISPs, because they do not participate actively either in the indexing system or in the resource exchange.

## 4 A USERS' PRIVACY MODULE FOR CLOSER

Section 3 focused on ISP needs, related to the circumscription of P2P traffic. Here we present CLOPS (CLOser Privacy Support), a privacy module for CLOSER that gives users an incentive to adopt CLOSER and hence to favor a wide spread of this new paradigm in the P2P community.

### 4.1 Rationale

The introduction of locality-awareness in P2P systems may clash with the indifference and the suspiciousness of users that, without proper incentives, are not motivated to adopt new P2P applications and paradigms. Furthermore, several publications [6], [7] demonstrate that locality-aware schemes may increase the download time, especially when peers are not uniformly distributed among the ISPs and their access bandwidths are heterogeneous. This leads to a *win-lose* situation for ISPs and users that further discourages users to adopt such systems. The oracle technique and P4P have an additional drawback from this perspective: users have to disclose information to ISPs, which are usually considered hostile [21], [22].

Hence, we need different incentives that, similarly to the download time, are of interest for users. Among the possible incentives, we select to focus on users' privacy, due to the effort that several user communities of software development are giving to the definition of *anonymous P2P systems* (e.g., ANts P2P [9], MUTE [10], OFF [11], Freenet [12]).

Using basic techniques, an eavesdropper that wants to compromise users' privacy can monitor their actions by (*i*) intercepting the control or data traffic generated by peers, (*ii*) acting as indexing system, by monitoring searches, shared resources, and downloads, and (*iii*) acting as a P2P user, by acquiring information during its apparently normal activity in the P2P overlay. The encryption features already deployed in modern P2P applications can easily prevent an eavesdropper to intercept P2P traffic and, consequently, this scenario is no longer interesting. Hence, we concentrate on the other privacy threats, which are of more interest in modern P2P systems.

The state of the art solutions concerning the users' behavior anonymity, also adopted in the abovementioned user-driven systems, is represented by [23] and [24], both proposed in 2002. These papers present two similar techniques based on the utilization of peers as proxy nodes, in conjunction with hard cryptography. A similar approach is also used in Tor [25], which has been specifically designed to anonymize TCP connections. These technologies make harder the connection tracing and, thus, hide who executes the requests, both when the eavesdropper controls the indexing system and when it acts as a normal P2P user. The penalty to pay when using these solutions is an increase of the download time [13], due to the utilization of possibly slow or overloaded intermediate proxy nodes to download resources whose size is generally large. This is perhaps the reason for which these techniques did not become widely popular. CLOPS, the users' privacy module we developed for CLOSER, avoids this issue by enabling direct downloads.

### 4.2 CLOPS overview

With respect to existing solutions, CLOPS achieves users' privacy by following a totally different approach: P2P applications automatically select and download resources, even if those are not requested by the user. This offers users' privacy because, observing a node behavior, it is hard to determine if resources were requested by the actual user or by an automatic download process. In particular, these automatic downloads can easily deceive an eavesdropper acting as a P2P user or indexing system as sharing or downloading a resource does not mean that the resource is shared or requested by the user.

In order to avoid penalizing actual traffic due to the consumption of precious access bandwidth of these additional downloads, it is necessary to introduce appropriate work-conserving scheduling algorithms that limit the CLOPS download rate. In particular, the downloading machine, which can discriminate among real and CLOPS downloads, gives higher priority to real down-

loads but ensures a minimum bandwidth guarantee to CLOPS downloads, selected as a small fraction of the total bandwidth available on the access link. In this way, CLOPS downloads can continue even when the access link is fully loaded, thus guaranteeing privacy, but do not penalize real downloads as in such a situation they consume a very small portion of the available bandwidth. The Class Based Queuing (CBQ) [26] algorithm can be used for this purpose as it is able to handle multiple classes at different priorities with minimum bandwidth guarantees. Hence, although possible alternatives exist, we propose to adopt the CBQ algorithm due to its proved effectiveness and wide adoption in many networking areas. Furthermore, several open-source CBQ implementations are available and can be seamlessly adapted to operate in the CLOSER context.

### 4.3 CLOPS content choice

Although from the privacy perspective CLOPS can select the resources to download in a random fashion, it could be worth investigating how the selection of such resources could influence the locality awareness of the system. In particular, techniques could be studied for favoring downloads of resources that may be of interest for the users of an ISP in a near future. In this way, users will be likely to download them from the inside of the ISP, thus improving traffic locality. However, our analytical and simulation results (presented in the following sections) shows how CLOSER by itself is able to keep local 98% of traffic, thus making any attempt to further investigate this aspect not very significant.

This considered, we just need to ensure that CLOPS downloads do not reduce the overall CLOSER performance. In fact, a completely random selection of resources clearly penalizes the locality properties of the system as resources may be downloaded from the outside of the ISP with high frequency. Hence, it is necessary to force CLOPS to download only a negligible percentage of resources placed outside the boundaries of the ISP. Let $p_{do}$ denote this percentage; a reasonable choice is $p_{do} = 0.1\% \div 1\%$[2]. This policy can be applied thanks to the localization information offered by CLOSER, which enables CLOPS to discriminate between resources placed inside or outside the boundaries of the ISP.

In order to be able to select a resource to download, CLOPS modules have to be aware of the resources available in the P2P system. This is obtained by deploying a gossip protocol that spreads among nodes the information about the existence of resources. In essence, whenever a resource request arrives at the indexing system, this includes in its reply the ID of some resources randomly selected among the ones it knows. Analogously, whenever an interaction occurs between two nodes to start a download, those share the IDs of a subset of the resources they know. This enables nodes to learn existing resources and hence perform CLOPS downloads.

2. Notice that $p_{do} = 0$ affects users' privacy as allows eavesdroppers to classify as real downloads the traffic exiting the ISP boundaries.

TABLE 2
Model notation

| Symbol | Meaning |
|---|---|
| $N$ | # of resources in the P2P system |
| $M$ | # of resources downloaded |
| $f(i)$ | Probability that a user requests a resource of popularity rank $i$ |
| size$(i)$ | Size of resource of poularity rank $i$ |
| $P_{\text{ISP}_j}$ | Prob. user belongs to the ISP$_j$ |
| $P$ | # of users in the P2P system |
| $\Omega$ | Average # of shared resources per user |
| $L$ | # of results obtainable by a real indexing system |

Appendix D, which can be found in the Supplementary File, describes a content encryption scheme that CLOPS adopts to avoid possible issues deriving from the presence of copyrighted or illegal material among the resources selected for automatic download. The appendix also details the algorithms adopted in a CLOPS-aware peer to perform both user-driven and automatic downloads.

## 5 A SIMPLE INTER-ISP TRAFFIC MODEL

Since inter-ISP links usually have the most significant associated cost, in this section we specifically focus on the performance of CLOSER in circumscribing P2P traffic within the ISP boundaries. In particular, we present a simple analytical model that shows how CLOSER outperforms not only the locality unaware systems (referred to as "LU" in the following), but also other locality-aware mechanisms (referred to as "ELA") in achieving traffic reduction on inter-ISP links, thus also demonstrating the importance of the sampled list problem in the locality-awareness context.

Since each P2P protocol adopts different parallel download strategies (e.g., BitTorrent clients simultaneously download different file chunks according to specific piece and peer selection policies) and we would like to investigate a general case, we do not consider parallel downloading in this model (analogously to the approach used in [3] for the oracle-based technique).

Due to space limitations, we present here the final outcomes of our analytical work. The complete analysis and some additional remarks are available in Appendix E, which can be found in the Supplementary File.

### 5.1 Traffic reduction on inter-ISP links

**CLOSER/LU reduction.** Given the notation described in Table 2, the percentage traffic reduction on inter-ISP links offered by CLOSER with respect to legacy systems can be obtained by

$$G_{\text{C/L}}\% \quad = \quad (1 - R_{\text{C/L}}) \cdot 100, \qquad (1)$$

where

TABLE 3
Traffic reduction on inter-ISP links.

| Scenario | Value |
|---|---|
| Closer/Locality Unaware gain | 97.9% |
| Closer/Existing Locality Aware gain | 94.65% |

$$R_{\mathrm{C/L}} = \frac{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\mathrm{ISP}_j}\right)^{f(i) \cdot P \cdot \Omega} \cdot \mathrm{size}(i)}{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\mathrm{ISP}_j}\right) \cdot \mathrm{size}(i)}.$$

**CLOSER/ELA reduction.** Analogously to the previous case, we have

$$G_{\mathrm{C/E}}\% = (1 - R_{\mathrm{C/E}}) \cdot 100, \quad (2)$$

where

$$R_{\mathrm{C/E}} = \frac{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\mathrm{ISP}_j}\right)^{f(i) \cdot P \cdot \Omega} \cdot \mathrm{size}(i)}{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\mathrm{ISP}_j}\right)^{L_R(i)} \cdot \mathrm{size}(i)}.$$

### 5.2 Traffic reduction evaluation

To quantify the real benefits of CLOSER in reducing the P2P traffic over inter-ISP links, we apply the above derived equations to a real-world case, adopting as a reference the network of Telecom Italia, a prominent Italian ISP. Data related to the Telecom Italia network that are of interest in this context are publicly available on the web. These are used to set the model parameters, as detailed in Appendix F, which can be found in the Supplementary File.

Under these assumptions, the percentage gain that CLOSER achieves with respect to both the traditional locality-unaware systems and the existing locality-aware solutions are reported in Table 3. We can observe how CLOSER guarantees about 98% gain with respect tu LU systems and about 94.5% gain with respect to ELA mechanisms. These results demonstrate the effectiveness of CLOSER in reducing the utilization of inter-ISP links, thus making it an interesting solution for ISPs to limit their operating costs.

This result is achievable because a significant percentage of traffic is generated by popular resources that, by definition, are provided by a large number of resource providers. This effect is totally unexploited by locality unaware system, while the existing locality systems efficiency is compromised by the sampled list problem discussed in the previous sections.

## 6 Simulation and Experimental Results

Simulations have been run to both validate the above presented analytical model and further evaluate the proposed architecture. Some background on our simulation study and the setting methodology for the several parameters involved are presented in Appendix G, which can be found in the Supplementary File. All results are presented with 95% confidence interval.

In addition to this simulation study, we developed a CLOSER-aware application to verify the feasibility of our solution. Appendix H, which can be found in the Supplementary File, describes this software module and presents some results obtained on PlanetLab.

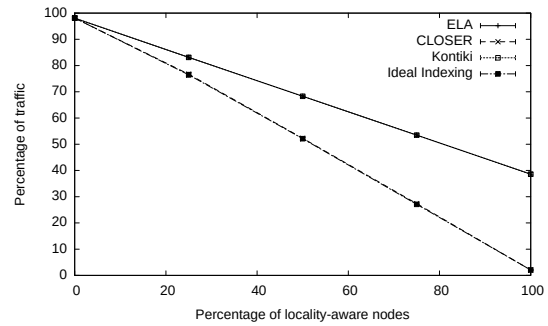


Fig. 1. Overall P2P traffic crossing the ISP borders

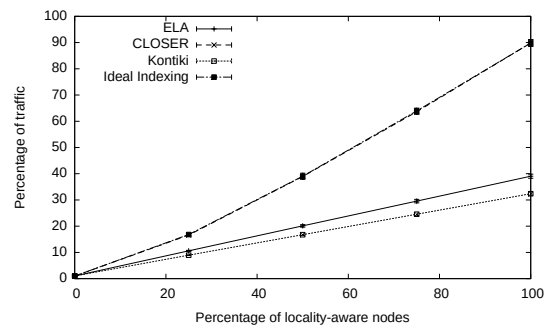


Fig. 2. Overall P2P traffic circumscribed to the requester's Area

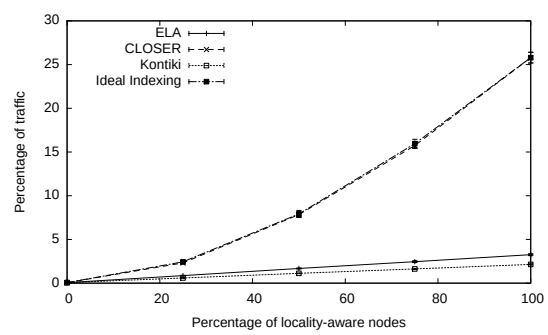


Fig. 3. Overall P2P traffic circumscribed to the requester's PoP

### 6.1 Bandwidth usage

A first set of simulations aims at identifying how the available link capacity is utilized. We do not consider CLOPS downloads, whose effects on traffic locality will be presented later in this section.

We evaluate the performance of: CLOSER, the newly described technique; LU, a generic legacy system without locality-awareness; Ideal Indexing, an ideal system that provides the whole list of content providers perfectly ordered according to the topological distance; ELA, the class of algorithms including Ono, the oracle, P4P, and ALTO; Kontiki, the simple mechanism used by Kontiki and described in Section 2. Kontiki and other ELA systems perform equal concerning the utilization of inter-ISP links, but they have to be handled separately in this simulation study as we also consider the circumscription of traffic within areas smaller than the entire ISP. In fact, Kontiki uses public IANA databases to acquire the localization information, which hence cannot be more specific than an AS number. Within the ISP boundaries, the resource provider selection of Kontiki is locality unaware, i.e., random.

Figure 1, Figure 2, and Figure 3 depict the link usage in different areas of the network as a function of the percentage of nodes in the P2P network adopting a locality-aware system. This is done to study the effects of a progressive adoption of locality-aware techniques.

Figure 1 reports on the usage of links with the tier 1 ISP. As expected, Kontiki and other ELA techniques have a similar behavior in this context (curves are overlapped in Figure 1). In fact, the effectiveness of both techniques is limited because of their ability of providing only a subset of the available resource providers (i.e., the sampled list problem). Both are outperformed by CLOSER, which mimics an ideal indexing system (again, curves are almost overlapped in the figure) thanks to its ability to offer the $L$ closest content providers, perfectly ordered according the topological distance. In fact, if a "local" resource provider exists, this will be included in the list and contacted by the querying user for downloading the file. If this peer is busy, the user will contact the next peer in the list, and so on until an available peer is found. Thereby, an ideal system providing a complete list of resource providers performs better than CLOSER only if more than $L$ local resource providers exist and all of them are busy at the same time, which is an event unlikely to occur. Notice that CLOSER outperforms the ELA architectures despite the latter require ISPs to deploy a powerful infrastructure composed by several servers. Table 4 compares the reduction of the inter-ISP link utilization obtained when the percentage of modified clients reaches 100% with the analytical results derived in Section 5.2, both confirming the effectiveness of CLOSER and validating our analytical model.

Figure 2 and Figure 3 show the amount of data that was circumscribed in an Area (the northern and southern Italy areas described above) and in a PoP, respectively. Also in these contexts CLOSER performs similar to an ideal indexing system, which confirms the effectiveness of the architecture also in handling the hierarchical localization information introduced in Appendix C.

TABLE 4
Comparison of Simulation and Analytical results.

| Scenario | Model | Simulation |
|---|---|---|
| $G_{E/L}$ | 60.74% | 60.66% |
| $G_{\mathrm{C/E}}$ | 94.65% | 94.68% |
| $G_{\mathrm{C/L}}$ | 97.90% | 97.91 % |

TABLE 5
Variation of inter-ISP link utilization due to CLOPS automatic downloads

| $p_{do}$ | Relative variation |
|---|---|
| 0.10 | -2.08 % $\pm$ 1.18 % |
| 0.25 | -1.84 % $\pm$ 1.19 % |
| 0.50 | -0.82 % $\pm$ 1.34 % |
| 0.75 | -0.21 % $\pm$ 1.18 % |
| 1.00 | 0.81 % $\pm$ 1.46 % |

### 6.2 CLOPS evaluation

To conclude our simulation study, we investigate the effects that CLOPS, the users' privacy module of CLOSER, has in the overall network performance. In particular, since CLOPS is based on automatic downloads, it is necessary to verify that this module does not affect the performance of CLOSER concerning the circumscription of traffic. Table 5 reports on the variation of inter-ISP link utilization due to the presence of CLOPS for different values of $p_{do}$ (i.e., the percentage amount of resources that CLOPS downloads from the outside of the ISP). Although one could expect a performance degradation equal in percentage to the adopted $p_{do}$ value, the table rather shows how we have a slight performance increase for small $p_{do}$ values and a slight decrease when $p_{do}$ grows. This is due to the presence, on average, of more copies of a resource within the boundaries of the ISP thanks to CLOPS downloads, which potentially lowers the utilization of inter-ISP links as reduces the probability for a user to download from the outside because internal providers are not available. However, since this event is unlikely to occur, CLOPS downloads results in a negligible increase of the system performance, especially when $p_{do}$ grows. Aside these considerations, we can conclude that small values of $p_{do}$ preserve user privacy and produce negligible effects on the utilization of inter-ISP links, which was our goal in this work.

The creation in the network of more copies of a given resource, due to CLOPS downloads, also explains the decrease of the average download time, although equal to 0.05%, we observed when CLOPS is used. As described in Section 4, a properly configured CBQ instance is introduced in the user machine to avoid penalizing real downloads due to CLOPS additional traffic (minimum bandwidth guaranteed to CLOPS download is fixed to 1% of the access bandwidth in these experiments). This considered, one probably expects an increase of the average download time, although slight thanks to the CBQ operation. Instead, the creation of more resource copies due to CLOPS increases the probability for a user to find a resource provider that is free and hence

actually available to upload the requested resource. This lowers the average time that users' downloads have to wait in resource providers internal queues before being allowed to actually start, and consequently it lowers the average download time. This download time reduction (as said above, 0.05% decrease with respect to the system operating without CLOPS) is negligible. However, our real purpose was to avoid increasing this time, which is actually achieved in our system.

## 7 Conclusions

This paper presents the Collaborative Locality-aware Overlay SERvice (CLOSER), an architecture that aims at lessening the usage of expensive international links in P2P file-sharing systems. This is obtained by exploiting traffic locality (i.e., a resource is downloaded from the inside of the ISP whenever possible) and generates significant cost savings for ISPs. Analytical and simulation results show the effectiveness of CLOSER, also with respect to other proposed techniques for traffic locality in P2P systems. Unlike other approaches, CLOSER can discriminate among all possible resource providers, thus avoiding the *sampled list problem*. This is obtained without transferring the complete list over the network, thus also preserving the scalability of the system.

CLOSER also introduces a privacy module as an incentive for users to switch to the new architecture. Furthermore, a CLOSER-aware application has been developed and described in the paper.

## References


[1] D. R. Choffnes and F. E. Bustamante, "Taming the torrent: a practical approach to reducing cross-isp traffic in peer-to-peer systems," in *Proceedings of the ACM SIGCOMM 2008 conference on Data communication*. Seattle, WA, USA: ACM, 2008, pp. 363–374.

[2] "Kontiki." [Online]. Available: http://www.kontiki.com

[3] V. Aggarwal, A. Feldmann, and C. Scheideler, "Can ISPS and P2P users cooperate for improved performance?" *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 3, pp. 29–40, 2007.

[4] H. Xie, Y. R. Yang, A. Krishnamurthy, Y. G. Liu, and A. Silberschatz, "P4P: provider portal for applications," in *Proceedings of the ACM SIGCOMM 2008 conference on Data communication*. Seattle, WA, USA: ACM, 2008, pp. 351–362.

[5] J. Seedorf, S. Kiesel, and M. Stiemerling, "Traffic localization for p2p-applications: The alto approach," in *Peer-to-Peer Computing, 2009. P2P '09. IEEE Ninth International Conference on*, sep. 2009, pp. 171 –177.

[6] W. Huang, C. Wu, and F. Lau, "The performance and locality tradeoff in bittorrent-like p2p file-sharing systems," in *Communications (ICC), 2010 IEEE International Conference on*, may. 2010, pp. 1 –5.

[7] F. Lehrieder, S. Oechsner, T. Hossfeld, Z. Despotovic, W. Kellerer, and M. Michel, "Can p2p-users benefit from locality-awareness?" in *Peer-to-Peer Computing (P2P), 2010 IEEE Tenth International Conference on*, aug. 2010, pp. 1 –9.

[8] R. Cuevas, N. Laoutaris, X. Yang, G. Siganos, and P. Rodriguez, "Deep diving into bittorrent locality," in *2010 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*. New York, NY, USA: ACM, 2010, pp. 349–350.

[9] "ANts P2P." [Online]. Available: http://antsp2p.sourceforge.net/

[10] "MUTE." [Online]. Available: http://mute-net.sourceforge.net/

[11] "The OFF system." [Online]. Available: http://offsystem.sourceforge.net/

[12] I. Clarke, S. G. Miller, T. W. Hong, O. Sandberg, and B. Wiley, "Protecting free expression online with freenet," *IEEE Internet Computing*, vol. 6, no. 1, pp. 40–49, 2002.

[13] P. Dhungel, M. Steiner, I. Rimac, V. Hilt, and K. Ross, "Waiting for anonymity: Understanding delays in the tor overlay," in *Peer-to-Peer Computing (P2P), 2010 IEEE Tenth International Conference on*, aug. 2010, pp. 1 –4.

[14] B. Liu, Y. Cui, Y. Lu, and Y. Xue, "Locality-awareness in bittorrent-like p2p applications," *Multimedia, IEEE Transactions on*, vol. 11, no. 3, pp. 361 –371, apr. 2009.

[15] M. Lin, J. Lui, and D.-M. Chiu, "An isp-friendly file distribution protocol: Analysis, design, and implementation," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 21, no. 9, pp. 1317 –1329, sep. 2010.

[16] P. Racz, S. Oechsner, and F. Lehrieder, "Bgp-based locality promotion for p2p applications," in *Computer Communications and Networks (ICCCN), 2010 Proceedings of 19th International Conference on*, aug. 2010, pp. 1 –8.

[17] H. Wang, J. Liu, B. Chen, K. Xu, and Z. Ma, "On tracker selection for peer-to-peer traffic locality," in *Peer-to-Peer Computing (P2P), 2010 IEEE Tenth International Conference on*, aug. 2010, pp. 1 –10.

[18] P. Maymounkov and D. Mazières, "Kademlia: A Peer-to-Peer information system based on the XOR metric," in *Peer-to-Peer Systems*, 2002, pp. 53–65.

[19] T. Klinberg and R. Manfredi, "Gnutella 0.6," Jun. 2002. [Online]. Available: http://groups.yahoo.com/group/the_gdf

[20] MaxMind LLC, "GeoIP." [Online]. Available: http://www.maxmind.com/

[21] E. Van Der Sar, "Uncovering the dark side of P4P," Aug. 2008. [Online]. Available: http://torrentfreak.com/uncovering-the-dark-side-of-p4p-080824/

[22] T. Mennecke, "Local sharing saves bandwidth on BitTorrent/P4P tests," Aug. 2008. [Online]. Available: http://www.slyck.com/story1748_Local_Sharing_Saves_Bandwidth_on_BitTorrentP4P_Tests

[23] M. J. Freedman and R. Morris, "Tarzan: a peer-to-peer anonymizing network layer," in *CCS '02: Proceedings of the 9th ACM conference on Computer and communications security*. New York, NY, USA: ACM, 2002, pp. 193–206.

[24] M. Rennhard and B. Plattner, "Introducing morphmix: peer-to-peer based anonymous internet usage with collusion detection," in *WPES '02: Proceedings of the 2002 ACM workshop on Privacy in the Electronic Society*. New York, NY, USA: ACM, 2002, pp. 91–102.

[25] "Tor: anonymity online." [Online]. Available: http://www.torproject.org

[26] S. Floyd and V. Jacobson, "Link-sharing and resource management models for packet networks," *Networking, IEEE/ACM Transactions on*, vol. 3, no. 4, pp. 365 –386, Aug. 1995.



**Marco Papa** is a Ph.D. student in Computer and System Engineering at the Department of Control and Computer Engineering of Politecnico di Torino (Technical University of Turin), Italy. He holds a B.S. Degree and M.S. Degree both in Computer Engineering. His research interests include quality of service, privacy and peer-to-peer technologies.

**Luigi Ciminiera** is professor of Computer Engineering at the Dipartimento di Automatica e Informatica of Politecnico di Torino, Italy. His research interests include grids and peer-to-peer networks, distributed software systems, and computer arithmetic. He is a coauthor of two international books and more than 100 contributions published in technical journals and conference proceedings. He is a member of the IEEE.

**Guido Marchetto** is a post-doctoral fellow at the Department of Control and Computer Engineering of Politecnico di Torino. He got his Ph.D. in Computer Engineering in April 2008 and his laurea degree in Telecommunications Engineering in April 2004, both from Politecnico di Torino. His research topics are peer-to-peer technologies, distributed services, and Voice over IP protocols. His interests include network protocols and network architectures.

**Fulvio Risso** is Assistant Professor at the Department of Control and Computer Engineering of Politecnico di Torino. He is author of several papers on quality of service, packet processing, network monitoring, and IPv6. Present research activity focuses on efficient packet processing, network analysis, network monitoring, and peer-to-peer overlays.

# Supplementary File

**CLOSER: A Collaborative Locality-aware Overlay SERvice**

Marco Papa Manzillo, Luigi Ciminiera, Guido Marchetto, Fulvio Risso

## Appendix A
## Additional literature review

Among the methods proposed to lower inter-ISP link utilization in P2P systems, a first solution that some ISPs adopted was to discriminate against P2P traffic, deliberately slowing or blocking it. This behavior reduced the Internet transit expenses and infrastructure upgrade costs, but collided with network neutrality principle and implied damage to the image of that ISPs, a loss of customers, and government investigations [1].

Other possible solutions rely on downloading resources, when possible, within the boundaries of the ISP, as originally proposed in [2]. A method is the deployment of caches for P2P traffic [3]–[5], so that there is a high probability for users to download popular content without crossing the ISP's boundaries. Another approach is based on intercepting and modifying P2P communications in order to redirect them to the closest peers providing the requested resource (referred to as *resource providers* in the following) [6], [7]. However, the former has a main drawback related to the possible onset of legal problems for ISPs due to copyright issues, which limits the deployment of this solution to closed systems where ISPs control the downloaded content. Additional complexities derive from the selection of resources to store and replace in caches. The latter does not have such drawbacks but it is impaired by the protocol encryption facilities provided by modern P2P applications. Even if there are techniques (e.g., [8]) that succeed in identifying such protocols, they cannot decipher the encrypted data and, thus, it becomes impractical to intercept and modify the exchanged information. The modification of one or more system components, discussed in Section 2, is a third approach belonging to such category and avoids the abovementioned issues.

## Appendix B
## Resource registration and retrieve procedures in CLOSER

Figure 1 and Figure 2 detail the resource registration and retrieve procedures of a CLOSER-aware peer and indexing system, respectively. Notice how peers communicate their location to the indexing system during the registration of a resource (procedure `PeerRegistration()` in Figure 1). The indexing system stores this information (procedure `Register()` in Figure 2) and can use it during resource lookups, when a requesting peer again communicates its own localization information (procedure `Request()` in Figure 1). In particular, the indexing system can sort all the query results as it already knows localization data and then communicate the closest $L$ resource providers to a querying user (procedure `Search()` in Figure 2).

```
proc PeerRegistration() ≡
  for Resource ∈ ResourcesToShare do
      IndexingSystem.Register(Resource.ID(), MyAddress,
                              MyLocation)
  od
end
proc Request(ResourceID) ≡
  Results = IndexingSystem.Search(ResourceID, MyLocation)
  // Download the resource and store it in Resource
  return Resource
end
```

Fig. 1. CLOSER Peer procedures

```
proc Register(ResourceID, PeerAddress, PeerLocation) ≡
  DB.Add(ResourceID, PeerAddress, PeerLocation)
end
proc Search(ResourceID, RequesterLocation) ≡
  AllResults = DB.Search(ResourceID)
  Sort(AllResults).ByMinDistanceFrom(RequesterLocation)
  Results = FirstLResults(AllResults)
  return Results
end
```

Fig. 2. CLOSER Indexing System procedures

## Appendix C
## Localization information in CLOSER

The definition of a proper localization information is essential for allowing CLOSER to effectively guarantee traffic segregation. The main objective is to limit the traffic within the boundaries of a given ISP in order to reduce the utilization of expensive inter-ISP links. However, the ISP itself could have the interest of confining the traffic in smaller ISP zones to avoid, for example, strategically important backbone links. For

*The authors are with the Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy (email: name.surname@polito.it).*

this reason, we define the localization information as a set of $k$ hierarchical localization data which give an arbitrarily detailed information about the location of a resource provider. The first identifier always represents a globally unique number. Similarly to other techniques, we select the AS Number (ASN) for this purpose[1]. Lower hierarchical levels can be chosen arbitrarily by each ISP to satisfy specific requirements. For example, besides the first globally unique ID, one can define three additional levels: (*i*) a country ID, (*ii*) a geographical area ID (e.g., the northeastern area of US), (*iii*) and a city or Point of Presence (PoP) ID, which together specify the location of a resource provider.

When the indexing system sorts the resource provider list during a resource lookup, it places the resource providers that have the same $k$-uple (e.g., are in the same ISP, the same country, in same city) of the requester on the top of the list, followed by the resource providers with the same most significant $(k-1)$-uple (e.g., the same ISP, the same country), and so on. This allows the traffic to be circumscribed within smaller areas identified by the localization data adopted.

## APPENDIX D
## CLOPS CONTENT ENCRYPTION

Although deceiving eavesdroppers, the operation of CLOPS could lead to a possible issue: while downloading resources with the aim of preserving the privacy of its user, the CLOPS module running on a host could retrieve and then share illegal content provided by malicious people. Since users may be imposed by law to constantly monitor what is shared by their P2P application, this could make these users punishable and, consequently, CLOPS unusable. This issue can be solved by encrypting all the resources shared in the P2P system and then forcing CLOPS to retrieve only the ciphered resources, without obtaining either the resource names or the decryption keys[2]. In this way, the material downloaded by CLOPS appears to the user just as pseudo random data and she has no way to reconstruct the original content. However, a privacy issue may arise because an eavesdropper acting as indexing system can discriminate between normal and CLOPS downloads as the former involve both the ciphered resource and the decryption key, while the latter involve only the ciphered resource. This can be avoided by borrowing the concept of proxy-based communications from existing solutions: whenever users interact with the indexing system (i.e., when publishing or locating resources), they use a traditional anonymity system based on proxies (e.g., [9], [10]),

1. We select a specific identifier to guarantee the interoperability among P2P applications developed by different communities or used in different ISPs. Among the possible choices, we select the ASN for its straightforward utilization. However, different identifiers can be adopted, conditioned to the achievement of an agreement among the involved parties.

2. Notice that the described procedure is different from using encrypted channels during peer communications, which is of little help to solve this issue.

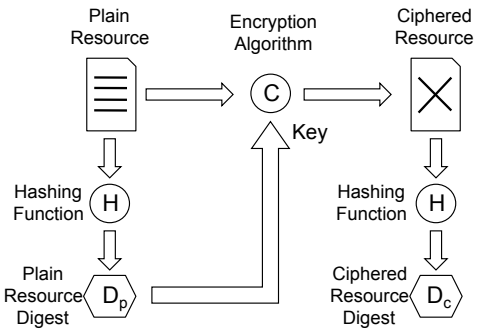


Fig. 3. Content encryption

so that the eavesdropper might discriminate between real and CLOPS downloads but it is hard to discover the origin of the requests. Notice that, since data exchanged with the indexing system generally does not require a high bandwidth because are relatively small, such policy does not affect the system performance. Downloads, which instead are bandwidth intensive, do not involve proxy nodes.

A simple method to obtain resource encryption is to let each user to randomly choose encryption keys. These have to be applied to the resources before their publication in the indexing system. However, the encryption key is preferable to be unique at content level (i.e., equal for resources containing the same content), so that the system performance increases as equal resources independently shared by different users can be part of the same swarming. CLOSER adopts the digest of the plain resource as encryption key, so that the same content result to be encrypted in the same way. Hence, whenever a user decides to share a resource, the P2P application computes the digest of the plain resource (denoted as $D_p$) and encrypts the resource using $D_p$ as encryption key, as shown in Figure 3. The P2P application also computes the digest of the ciphered resource (denoted as $D_c$), which is adopted as resource ID. $D_p$ and $D_c$ are then published in the indexing system to let users to acquire these two indexes through keyword-based searches. $D_c$ is used to retrieve the encrypted resource, while $D_p$ is used to decrypt the file.

On the contrary, the gossip mechanism described in Section 4.3 only spreads the information about the existence of undefined resources identified by different $D_c$ values, without providing either the full name or the associated $D_p$. The CLOPS module can use the obtained IDs $D_c$ to retrieve the correspondent ciphered resources, which however cannot be either decrypted or identified as there is no way to obtain either the key $D_p$ or the resource name. Hence, the user should not be considered responsible for the content downloaded by the CLOPS process as she cannot either choose the resources to download or even know what they are. Notice that the downloaded resources can however collaborate in improving the swarming as each node can register itself

```
proc Request(Filename) ≡
  CipheredDigest, PlainDigest =
      IndexingSystem.RetrieveIDByFilename(Filename)
  SourceResults =
      IndexingSystem.Search(CipheredDigest, MyLocation)
  // Download the resource and store it in CipheredResource
  Resource = Decrypt(CipheredResource).WithKey(PlainDigest)
  return Resource
end
proc CLOPSAutomaticRequest() ≡
  CipheredResourceID = SelectFrom(GossipKnownIDs)
  SourceResults =
      IndexingSystem.Search(CipheredResourceID, MyLocation)
  // Download the resource and store it in CipheredResource
  return CipheredResource
end
```

Fig. 4. Request procedures in a CLOPS-aware node

in the indexing system as a provider of the resource whose ID is the particular value of $D_c$.

Figure 4 details the algorithms adopted in a CLOPS-aware peer to perform both a resource download actually requested by a user and a CLOPS automatic download. Procedure `Request()` refers to the real downloads performed by users. Whenever they run a "by keyword" search (method `RetrieveIDByFilename()`, offered by the indexing system), they retrieve both the resource ID (i.e., *CipheredDigest*) and the decryption key (i.e., *PlainDigest*). Then, they perform a `Search()` to download the ciphered resource, which they can decrypt and use. Notice instead how CLOPS automatic downloads randomly select a resource ID among the *GossipKnownIDs* and then do not retrieve the decryption key, which makes the user unable to decrypt the downloaded content.

## APPENDIX E
## INTER-ISP TRAFFIC MODEL: COMPLETE ANALYSIS

We start the analysis by evaluating the fraction of P2P traffic traversing inter-ISP links in the three cases under examination (i.e., locality-unaware systems, existing locality-aware systems, and CLOSER). Then, we use these results to compare the considered techniques in terms of traffic reduction over inter-ISP links.

### E.1 Fraction of P2P traffic traversing the ISP boundaries

#### *E.1.1 Locality-unaware*

The download process of a legacy locality-unaware system, which will be referred to as "LU" in the following, can be summarized in three steps: (*i*) the requester queries the indexing system for a given resource; (*ii*) the indexing system returns a random subset of up to $L$ resource providers (by default, $L = 50$ in BitTorrent); (*iii*) the requester downloads from a resource provider selected in an almost random fashion (i.e., the first node that allows it to download) among the retrieved $L$ peers. Thus, the probability for the download to be circumscribed within one single ISP is equal to the probability for a generic node to belong to the requester ISP. We can argue that the fraction of data received from another ISP in the LU scenario is equal to:

$$\begin{aligned} F_{LU} &= \frac{M \cdot \sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\text{ISP}_j}\right) \cdot \text{size}(i)}{M \cdot \sum_{i=1}^{N} s(i) \cdot f(i) \cdot \text{size}(i)} \\ &= \frac{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\text{ISP}_j}\right) \cdot \text{size}(i)}{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \text{size}(i)}, \end{aligned} \quad (1)$$

where

$$s(i) = \begin{cases} 1 & \text{if } f(i) \cdot P \cdot \Omega \geq 1 \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

Since resources downloaded by users are then generally shared to improve the swarming performance, $f(i)$ also models with reasonable accuracy the probability for a user to share a resource of popularity rank $i$. Moreover, $P \cdot \Omega$ is the number of resources shared globally and $f(i) \cdot P \cdot \Omega$ represents the number of copies of the $i$-th resource available in the system. The function $s(i)$ is introduced because, if there is less than one copy of the $i$-th resource in the system, that resource cannot be downloaded and, thus, does not generate traffic. Usually, users do not download twice the same resource and, thus, (2) may seem imprecise since the $\Omega$ resources shared by each node are not independent. However, [4] describes that a Zipf-Mandelbrot distribution properly models $f(i)$ and shows how such distribution takes into account this issue.

#### *E.1.2 Existing locality-aware systems*

As explained in Section 2, the list of resource providers that in real scenarios the oracle, P4P, and ALTO enabled nodes obtain from the indexing system is generated according to the same principles, and thus results comparable in these three cases. Such list is also the best result achievable by Ono and Kontiki in ideal conditions — i.e., when the localization information is adequately accurate. Thus, the oracle, P4P, ALTO and an ideal instance of Ono and Kontiki achieve the same results from a traffic distribution point of view. In the following we represent these technologies as "Existing locality-aware" (ELA) systems. In these contexts, as in the legacy case, the indexing system returns a random subset of up to $L$ results but the requester or the oracle/iTracker facility sorts the list and, thus, the node chosen among the $L$ available is not random. Assuming that resource providers are available for upload, the data fraction received through the inter-ISP links is equal to:

$$\begin{aligned} F_{ELA} &= \frac{M \cdot \sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\text{ISP}_j}\right)^{L_R(i)} \cdot \text{size}(i)}{M \cdot \sum_{i=1}^{N} s(i) \cdot f(i) \cdot \text{size}(i)} \\ &= \frac{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\text{ISP}_j}\right)^{L_R(i)} \cdot \text{size}(i)}{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \text{size}(i)}, \end{aligned} \quad (3)$$

where

$$L_R(i) = min(L, f(i) \cdot P \cdot \Omega). \quad (4)$$

Notice that (1) and (3) are similar; the main difference is that, in the ELA context, the requester fetches the resource from a node belonging to another ISP only if none of the $L_R$ results belongs to its ISP. The function $L_R(i)$ is used in place of $L$ because the indexing system could be unable to return $L$ resource providers if there are no enough results.

#### E.1.3 CLOSER

In this context, the download process is different with respect to the previous scenarios and follows these steps: (*i*) the requester queries the CLOSER indexing system for the desired resource; (*ii*) the CLOSER indexing system returns a subset of up to $L$ results. If there are resources in the same ISP of the requester, it is guaranteed that they are among the results. Thus, in the CLOSER context, the requester downloads from a node belonging to another ISP only if none of the resource providers in the whole system belongs to its ISP. The CLOSER locality-aware mechanism is denoted as "CLO" in the following equations. The fraction of data received through the inter-IPS links can be expressed as:

$$\begin{aligned} F_{\text{CLO}} &= \frac{M \cdot \sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\text{ISP}_j}\right)^{f(i) \cdot P \cdot \Omega} \cdot \text{size}(i)}{M \cdot \sum_{i=1}^{N} s(i) \cdot f(i) \cdot \text{size}(i)} \\ &= \frac{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\text{ISP}_j}\right)^{f(i) \cdot P \cdot \Omega} \cdot \text{size}(i)}{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \text{size}(i)}. \quad (5) \end{aligned}$$

### E.2 Traffic reduction on inter-ISP links

We use now equations previously derived to evaluate the traffic reduction that CLOSER achieves on inter-ISP links.

#### E.2.1 CLOSER/LU reduction

The percentage traffic reduction on inter-ISP links offered by CLOSER with respect to legacy systems can be obtained by

$$\begin{aligned} G_{\text{C/L}}\% &= \left(\frac{F_{LU} - F_{CLO}}{F_{LU}}\right) \cdot 100 = \\ &= \left(1 - \frac{F_{CLO}}{F_{LU}}\right) \cdot 100 = \\ &= (1 - R_{\text{C/L}}) \cdot 100, \quad (6) \end{aligned}$$

where $R_{\text{C/L}}$ can be evaluated as

$$R_{\text{C/L}} = \frac{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\text{ISP}_j}\right)^{f(i) \cdot P \cdot \Omega} \cdot \text{size}(i)}{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\text{ISP}_j}\right) \cdot \text{size}(i)}. \quad (7)$$

For a given $\text{ISP}_j$, $P_{\text{ISP}_j} \in (0,1)$[3]. Hence, $\left(1 - P_{\text{ISP}_j}\right) \in (0,1)$, where $\left(1 - P_{\text{ISP}_j}\right)$ is the probability for a user *not* to belong to $\text{ISP}_j$. Furthermore, $f(i) \cdot P \cdot \Omega \geq 1$, $\forall i$ as at least one copy of a resource has to be present in the network to have a download request for that resource. Since $a^x = b$, $b \in (0, a)$ when $a \in (0,1)$, $x \geq 1$, we can argue that $R_{\text{C/L}} \leq 1$ and, consequently, $G_{\text{C/L}}\% \geq 0$. Furthermore, since popular resources are generally shared by a large number of resource providers, $f(i) \cdot P \cdot \Omega \gg 1$ for these resources, which represent the most significant contribution in (7). This considered, we can argue that CLOSER offers significant bandwidth savings on inter-ISP links with respect to the traditional LU techniques. Furthermore, the higher the number of resource providers $f(i) \cdot P \cdot \Omega, \forall i$, the higher this gain.

3. By definition of probability, $P_{\text{ISP}_j} \in [0,1]$. However, it is reasonable to assume that, for actual ISPs, $P_{\text{ISP}_j} \in (0,1)$ as a probability equal to one would assume a single ISP in the whole Internet, while a probability equal to zero would assume an ISP without users.

TABLE 1
Reference Values

| Symbol | Reference Value |
|---|---|
| $N$ | 633,106 |
| $f(i)$ | Zipf-Mandelbrot $N, q = 25, \alpha = 0.55$ |
| $P_{\text{ISP}_{TI}}$ | 1.89% |
| $P$ | 2,041,590 |
| $\Omega$ | 84 |
| $L$ | 50 |

#### E.2.2 CLOSER/ELA reduction

Analogously to the previous case, we have

$$G_{\text{C/E}}\% = (1 - R_{\text{C/E}}) \cdot 100, \quad (8)$$

where

$$R_{\text{C/E}} = \frac{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\text{ISP}_j}\right)^{f(i) \cdot P \cdot \Omega} \cdot \text{size}(i)}{\sum_{i=1}^{N} s(i) \cdot f(i) \cdot \left(1 - P_{\text{ISP}_j}\right)^{L_R(i)} \cdot \text{size}(i)}. \quad (9)$$

From (4), $L_R(i) = f(i) \cdot P \cdot \Omega$ if $f(i) \cdot P \cdot \Omega \leq L$. Hence, ELA systems and CLOSER perform equally until, for each resource, the number of available copies is lower than the maximum amount of results that ELA systems can offer to a querying user. Then, the sampled list problem comes into play and CLOSER starts to offer larger inter-ISP bandwidth savings than ELA systems. In conclusion, the lower the number of results $L$ a user can obtain from the indexing system, the higher is the gain that CLOSER offers on ELA systems. Furthermore, given a value for $L$, this gain increases with the number of resource copies present in the network.

## Appendix F
## Inter-ISP traffic model: parameter setting

We evaluated $P_{\text{ISP}_{TI}}$, i.e., the probability for a P2P user to be a Telecom Italia customer, as follows. Assuming the number of P2P users to be directly proportional to the number of ISP customers, we obtained the distribution of users among countries from the Internet World Stats

website [11], which points out how the percentage of the Italian Internet population was about 2.51% at August 2009. Furthermore, the percentage of Telecom Italia customers among the total Italian Internet population is about 75%, derived by comparing the ISP investor relation [12], which includes the number of broadband Internet subscriptions, with the total Italian broadband Internet subscriptions provided by the Italian National Statistical Institute (ISTAT) [13]. We can conclude that $P_{\text{ISP}_{TI}} = 1.89\%$ in our case-study.

Concerning the request distribution $f(i)$, we adopt a Zipf-Mandelbrot distribution, as proposed in [14]. Clearly, this choice does not keep into account the CLOPS automatic downloads. However, the portion of CLOPS traffic traversing inter-ISP links can be considered negligible as resources to download are chosen within the ISP boundaries with very high probability (see Section 4.3). This is also confirmed by the simulation results presented in Section 6. Among the other sets of parameters defined in [14], we adopt the ones that the authors derived for the AS 2609 considered in the paper, which belongs to a mainly residential ISP and, consequently, can be considered applicable in the Telecom Italia scenario.

Other parameters are set according to the following principles and summarized in Table 1: $\Omega$ is the average value of a Weibull distribution, as presented in [15]; $N$ is the value observed by Gummadi et al. [16]; $P$ is the average number of active users in the eDonkey servers [17] listed in the full list available at [18] and excluding the fake server using the Bluetack level1 ipfilter [19]. Although these systems can operate with any file-sharing protocol, eDonkey is considered as a case study for both the large amount of publicly available data and the simplicity of the protocol, which offers the opportunity to easily derive missing data — e.g., the just mentioned number of users $P$.

## APPENDIX G
## SIMULATION STUDY: BACKGROUND AND PARAMETER SETTING

A first issue to address in our simulation study was the selection of a proper network simulator which complies with the requirements of our study. Existing packet-level simulators have been discarded because of their resource-hungry nature. On the contrary, application level simulators generally reduce the simulated events to the node message exchange and consider only trivial (or do not consider at all) underlying topologies. These reasons drove us to develop an ad-hoc simulator. The simulator is written in C++, is composed of about 10,000 lines of code, and models the network layer at connection level, thus resulting adequate for the measurement of the bandwidth usage over links that we require in this context. Furthermore, the program has been developed for simulating a large number of nodes. In presence of bottleneck links, the bandwidth is equally shared among the connections, while the absence of routing loop guarantees that the bandwidth-sharing algorithm terminates in a finite time. Analogously to the analytical model presented in Section 5, we do not consider parallel downloads in order to obtain more general results.

Concerning the CLOSER localization data, we adopt a three-level information, which we consider a reasonable choice for a significant number of ISPs, providing enough information about the location without making the system cumbersome. In particular, the levels represent: (*i*) an ISP unique ID (using the ASN), (*ii*) an area ID, i.e., a sub zone of the ISP WAN, and (*iii*) a Point of Presence (PoP) ID. The simulation time is set to 20 days.

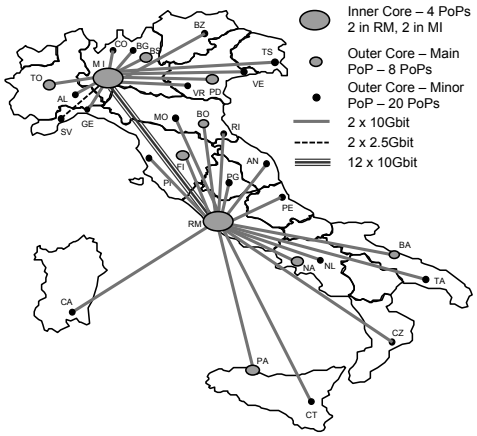


Fig. 5. Telecom Italia network topology

Telecom Italia, the prominent Italian ISP adopted as a reference in Section 5.2, is considered as a case study also in this section. In particular, the adopted topology reproduces the real structure of the Telecom Italia network [20] (see Figure 5), represented at PoP granularity. The internal network is divided in two areas: the first one that covers northern Italy and the second one that covers the central and southern Italy. PoPs are almost equally divided between the northern and the southern areas. This topology is typical also of many ISPs in Europe. Other ISPs are represented as a unique area connected with the Telecom Italia network through an inter-ISP link dimensioned as the link between Telecom Italia and its tier 1 provider (i.e., 200 Gbps [20], properly reduced to meet some constraints that will be explained in the following).

Analogously to what applied in Section 5.2, we assume the number of P2P users in our Italian ISP to be directly proportional to the number of ISP customers, which, as derived in Appendix F, represents about 1.89% of the total Internet population. Furthermore, we assume the topological distribution of P2P users among the ISP PoPs to be directly proportional to the geographical distribution of the ISP customers: each PoP serves all the population in the town where the PoP is located, while the remaining population of a region is equally divided among the PoPs located in that region. Regions

without PoPs are aggregated to the topological closest region with at least one PoP. This topology allows us to accurately reproduce the download traffic pattern of the ISP users. Access bandwidths are considered uniform for simplicity and equal to 7 Mbps (download) and 400 Kbps (upload), which are the values characterizing a large percentage of Telecom Italia users.

We adopt a Poisson distribution for both the session lengths and the user arrivals. We set the average session length to 40 hours, according to what reported in [15]. Furthermore, we set the average arrival rate to 1 user per second, which coupled with the above session length results in an average overlay population of 144,000 nodes in the steady state. This overlay size guarantees the significance of the obtained results and meets our memory and CPU constraints. Since the overlay population adopted in the analytical model presented in Section 5 is 2,041,590 nodes (As described in Appendix F, we considered the average population of the eDonkey network for simplicity), while the simulated average population is instead 144,000 nodes, the capacity of the backbone links has been proportionally reduced with respect to the original values depicted in Figure **??**. These capacities have been further reduced by 63% since Schulze et al. [21] measured that eDonkey users are about the 37% of the total Internet population and the simulator does not reproduce additional traffic for simplicity.

The number of resources shared by users, represented by $\Omega$ in the analytical model presented in Section 5, is generated by adopting a Weibull distribution with scale $= 42$ and shape $= 0.5$, as depicted in [15]. The selection of the resources to share or request is modeled with a Zipf-Mandelbrot distribution, as proposed by in [14]. Analogously to analytical model, we adopt the Zipf-Mandelbrot parameters measured for the AS 2609 of the paper since it belongs to a mainly residential ISP, and, thus, applies also in our case. Furthermore, we assume the resources to be uniformly distributed among ISPs.

We also need to derive the number of unique resources present in the system ($N$ in the analytical model), which is highly related to the number of users and hence needs to be rescaled to fit with the overlay population adopted in this simulation. In the following we assume the number of unique resources to be directly proportional to the number of users, thus obtaining a number of unique resources equal to 44,655 by linearly rescaling the base value of $N$ reported in Appendix F. Notice how this choice represents a worst-case setting as the Zipf-Mandelbrot distribution, which models the probability for a resource to be shared in the P2P system, is a power-low distribution and, thus, is long-tailed. In fact, reducing the number of total users, the amount of resources for which the number of providers would result to be less than one significantly increases; consequently, the number of unique resources decreases more than linearly with the number of users.

The distribution of the time elapsing between resource download requests is obtained empirically after analyzing the eDonkey traffic coming from/to the network of the University campus. This traffic analysis lasted two weeks on a population of about 4,500 users. Download requests resulted to fit with a Poisson process with rate $\lambda \approx 6.75 \cdot 10^{-2}$ downloads/s. As said above, this rate has been measured for a system composed of about 4,500 users, but we simulate an average number of 144,000 users. Hence, by exploiting the statistical properties of Poisson processes, we assume our scenario as the union of 32 systems composed by 4,500 users each. Thus, we model the download request arrivals with a Poisson process with rate $\lambda_j = 2.16$ downloads/s.

## Appendix H
## A CLOSER-aware P2P application

A CLOSER-aware application has been developed. It has been derived from the well known C++ aMule client [22] and uses the Kad DHT as CLOSER indexing system. The original client already includes the Kad DHT, which we have duplicated and modified in order to enable the application to perform both locality-aware and traditional searches. Concerning the acquisition of the locality information, we were unable to both obtain the cooperation of different ISPs and define a standard procedure to interact with them. Hence, we developed a module that retrieves the node localization information from the distributed database GeoIP [23]. Alternatively, the localization information may be manually configured in the system, which may be useful for debugging and experimenting purposes. This information is sent to the indexing system during both resource registrations and searches, as described in Section 3.

The utilization of this application for validation purposes is not possible for scalability reasons. In fact, it is impossible to involve and remotely handle a large number of nodes located worldwide, as well as it is not trivial to generate real P2P traffic over these nodes. Rather, this experimental work aims at proving the real feasibility of CLOSER, showing how the locality-aware features can be seamlessly introduced in an existing client. An extensive validation of CLOSER is instead obtained by analysis and simulation, as described in Section 5 and Section 6. Given the purpose of this experimental work, it is worth noticing that the above described modifications required the update of about 1,650 lines of code, the deletion of about 900 lines of code and the insertion of about 2,300 lines of code. If compared to the total number of lines of the aMule client (about 150,000), these data confirm the feasibility of the proposed approach.

However, some experiments are run over the PlanetLab infrastructure for the sake of completeness. Besides the low overall number of nodes, the PlanetLab network has another limitation: PlanetLab nodes are distributed so that each ISP contains only a few of them (about two in many cases). This significantly reduces the probability for a download to involve a "local" node, thus

TABLE 2
List of resource providers obtained by the node planetlab1lannion.elibel.fr, located in AS 1, Area 1, PoP 1

| Without CLOSER | | With CLOSER | |
|---|---|---|---|
| Provider IP address | Location | Provider IP address | Location |
| 130.192.157.132 | 1.1.2 | 192.43.193.71 | 1.1.1 |
| 87.84.153.115 | 2.2.3 | 130.192.157.132 | 1.1.2 |
| 128.195.54.161 | 1.2.1 | 128.135.11.152 | 1.1.4 |
| 144.206.66.58 | 1.3.1 | 128.195.54.161 | 1.2.1 |
| 198.175.112.105 | 2.2.4 | 144.206.66.58 | 1.3.1 |
| 192.43.193.71 | 1.1.1 | 213.131.1.102 | 2.2.2 |
| 213.131.1.102 | 2.2.2 | 87.84.153.115 | 2.2.3 |
| 128.135.11.152 | 1.1.4 | 198.175.112.105 | 2.2.4 |

making a locality-awareness almost useless. Hence, we arbitrarily defined a logical topology over the PlanetLab network consisting of two ASs, each divided in two Areas. Each Area is further divided in four PoPs. We defined a 3-layer localization information, consisting of an AS ID, an Area ID, and a PoP ID, which we manually configured on each peer exploiting the abovementioned configuration feature we included in our application. Table 2 shows the list of resource providers obtained by our modified client running on the PlanetLab's machine "planetlab1lannion.elibel.fr" located in AS 1, Area 1, PoP 1. In the table, we introduce the notation $(x.y.z)$ for the localization information, where $x$ represents the AS ID, $y$ the Area ID, and $z$ the PoP ID. Notice how the list is perfectly ordered when CLOSER is used. This experiment verifies the proper operation of our CLOSER implementation. Future work will be dedicated to further experimenting with our application, possibly involving a larger number of nodes.

## REFERENCES


[1] E. Bangeman, "FCC officially opens proceeding on comcast's P2P throttling," *Ars Technica*, 1 2008. [Online]. Available: http://arstechnica.com/tech-policy/news/2008/01/fcc-officially-opens-proceeding-on-comcasts-p2p-throttling.ars

[2] T. Karagiannis, P. Rodriguez, and K. Papagiannaki, "Should internet service providers fear peer-assisted content distribution?" in *Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*. Berkeley, CA: USENIX Association, 2005.

[3] M. Hefeeda and O. Saleh, "Traffic modeling and proportional partial caching for peer-to-peer systems," *Networking, IEEE/ACM Transactions on*, vol. 16, no. 6, pp. 1447 –1460, dec. 2008.

[4] M. Hefeeda and B. Noorizadeh, "On the benefits of cooperative proxy caching for peer-to-peer traffic," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 21, no. 7, pp. 998 –1010, jul. 2010.

[5] F. Lehrieder, G. Dan, T. Hossfeld, S. Oechsner, and V. Singeorzan, "The impact of caching on bittorrent-like peer-to-peer systems," in *Peer-to-Peer Computing (P2P), 2010 IEEE Tenth International Conference on*, aug. 2010, pp. 1 –10.

[6] R. Bindal, P. Cao, W. Chan, J. Medved, G. Suwala, T. Bates, and A. Zhang, "Improving traffic locality in BitTorrent via biased neighbor selection," in *Distributed Computing Systems, 2006. ICDCS 2006. 26th IEEE International Conference on*, 2006.

[7] S. James and P. Crowley, "Imp: Isp-managed p2p," in *Peer-to-Peer Computing (P2P), 2010 IEEE Tenth International Conference on*, aug. 2010, pp. 1 –9.

[8] M. Dusi, M. Crotti, F. Gringoli, and L. Salgarelli, "Tunnel hunter: Detecting application-layer tunnels with statistical fingerprinting," *Comput. Netw.*, vol. 53, no. 1, pp. 81–97, 2009.

[9] M. J. Freedman and R. Morris, "Tarzan: a peer-to-peer anonymizing network layer," in *CCS '02: Proceedings of the 9th ACM conference on Computer and communications security*. New York, NY, USA: ACM, 2002, pp. 193–206.

[10] M. Rennhard and B. Plattner, "Introducing morphmix: peer-to-peer based anonymous internet usage with collusion detection," in *WPES '02: Proceedings of the 2002 ACM workshop on Privacy in the Electronic Society*. New York, NY, USA: ACM, 2002, pp. 91–102.

[11] Miniwatts Marketing Group, "Internet world stats," Feb. 2009. [Online]. Available: http://www.internetworldstats.com

[12] Telecom Italia S.p.A., "Resoconto intermedio di gestione al 30 settembre 2008." [Online]. Available: http://www.telecomitalia.it/TIPortale/docs/investor/html/trimestrale0809/download/TI-2008-terzo-trimestre.pdf

[13] Istituto Nazionale di Statistica (Istat), "Le imprese di telecomunicazioni (italian)," Dec. 2008. [Online]. Available: http://www.istat.it/salastampa/comunicati/non_calendario/20081201_01/

[14] O. Saleh and M. Hefeeda, "Modeling and caching of Peer-to-Peer traffic," in *Network Protocols, 2006. ICNP '06. Proceedings of the 2006 14th IEEE International Conference on*, 2006, pp. 249–258.

[15] V. Aggarwal, O. Akonjang, A. Feldmann, R. Tashev, and S. Mohr, "Reflecting P2P user behaviour models in a simulation environment," in *Parallel, Distributed and Network-Based Processing, 2008. PDP 2008. 16th Euromicro Conference on*, 2008, pp. 516–523.

[16] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan, "Measurement, modeling, and analysis of a peer-to-peer file-sharing workload," *SIGOPS Oper. Syst. Rev.*, vol. 37, no. 5, pp. 314–329, 2003.

[17] A. Klimkin, "eDonkey protocol specification v0.6.2," 2003. [Online]. Available: http://garr.dl.sourceforge.net/sourceforge/pdonkey/eDonkey-protocol-0.6.2.html

[18] "Server list for edonkey and emule." [Online]. Available: http://ed2k.2x4u.de

[19] "Bluetack internet security solutions (B.I.S.S.)." [Online]. Available: http://www.bluetack.co.uk

[20] A. Soldati, "Telecom italia ip backbone and peering policies," in *Italian Peering Forum (PFI 2008)*, 2008.

[21] H. Schulze and K. Mochalski, "Internet study 2008/2009," ipoque GmbH, Tech. Rep., 2009. [Online]. Available: http://www.ipoque.com/resources/internet-studies/internet-study-2008_2009

[22] "aMule." [Online]. Available: http://www.amule.org/

[23] MaxMind LLC, "GeoIP." [Online]. Available: http://www.maxmind.com/