Enriched Network-aware Video Services over Internet Overlay Networks

www.envision-project.org

Deliverable D4.3

Final Specification of Consolidated Overlay View, Data Management Infrastructure, Resource Optimisation and Content Distribution Functions

Public report, Version 1, 2 August 2012

Authors

UCL Eleni Mykoniati, Raul Landa, Lawrence Latif, Miguel Rio, David Griffin

ALUD Ivica Rimac, Klaus Satzke, Nico Schwan

LaBRI

TID Nikolaos Laoutaris, Oriol Ribera Prats

LIVEU

Reviewers Bertrand Mathieu, Noam Amram

Abstract This document specifies the final refinements of the network-aware overlay application techniques built by the ENVISION project. The tradeoff between application optimality and ISP cost is modelled and techniques for consolidating preferences across different ISPs are specified. A hybrid gossip and structured n-casting protocol is proposed for the exchange of resource availability information following the desirability of the resources based on some selection criteria. The deliverable elaborates on a distribution tree optimisation procedure, designed to enable interactive video applications to meet their strict delay requirements by complementing their participant resources with High-Capacity Nodes provided as a service by the network operators in strategic places. Finally, a content distribution optimisation technique is presented that plans the content transmission based on network information about the ISP cost over time and application information for predicting the content demand at particular locations and times.

Keywords Distributed Resource Data Management, Anycast, Manycast, Hierarchical Clustering, Hash Sketch, Network-aware Content Distribution, Interactive Video, Overlay Application, Tree Optimisation Algorithm, High-Capacity Node

Project funded by the European Union under the
Information and Communication Technologies FP7 Cooperation Programme
Grant Agreement number 248565

EXECUTIVE SUMMARY

This document is the third and final WP4 deliverable of the ENVISION project. The project advocates the cross-layer optimisation between network and application overlay functions through the Collaboration Interface between Network and Applications (CINA), documented in [D3.3]. This deliverable complements [D4.2] and summarises the final refinements of the developed network-aware overlay application techniques.

The information provided by the ISPs through the CINA interface may reflect the preferences of the ISPs determined based on their particular business policies and optimisation objectives. In the second case, a model for the tradeoff between application optimality and ISP costs is proposed and can be used to study the design choices of overlay applications and the related possibilities for collaboration with the underlying ISPs. Based on the insights gained by the work on ISP preferences in WP3, voting schemes are proposed for resolving incompatibilities between the preferences of different ISPs.

One of the challenges of a dynamic and large scale overlay network, is to maintain an accurate view of its participating nodes, application resources and their current engagement in the distribution of content items, in a way that it scales with the size of the overlay and is responsive to a large number of queries. To address this problem, an n-casting system is proposed, whereby a querying overlay node can discover its closest, in terms of network proximity, n distinct resources performing a particular overlay function. N-casting builds a distributed indexing protocol on top of an hierarchical clustering of the network endpoints and uses a statistical data structure for the efficient compression of the routing information.

For a tree-based streaming system developed in the context of interactive video applications, the limited capacity provided by the user end systems (peers) may result in the construction of a delivery tree violating the latency upper bound requirement associated with interactive video. One way to mitigate this problem is to introduce nodes in the overlay tree with higher fan-out degrees using the High Capacity Node network service provided by the network operators as described in [D3.3]. A Tree Optimisation algorithm designed to provide a structured approach to this problem is proposed. In the target scenario, participants join the overlay and form the end-system distribution tree. Once the latency threshold has been violated, the algorithm is triggered to optimise the distribution tree in a cost-effective manner by including of High Capacity Nodes in strategic locations.

The increasing popularity of user-generated content and the rise of online social networks as a distribution mechanism has increased the demand for long-tailed content, i.e. content that is popular among small groups of users. TailGate is a content distribution optimisation technique that plans the content transmission to a particular destination based on network information about the cost of using that link over time and application information for predicting the content demand at particular locations and times.

This deliverable concludes the design and specifications work in WP4. The techniques specified here and in previous deliverables will be evaluated in WP6, and the final evaluation results will be published in D6.2 at the end of the year.

TABLE OF CONTENTS

1. INTRODUCTION

In WP4, the focus is on developing techniques for enabling high-volume future media applications to be distributed over large and dynamic overlay networks operating in collaboration with the underlying ISPs. To this end, a number of techniques are developed, including supporting functions such as the consolidation of the information received by the ISPs and the scalable management of information about the overlay resources, and content distribution optimisation functions tailored to specific applications making efficient use of the underlying network capabilities provided at each ISP.

The major part of the work has been concluded and described in detail in previous deliverables, including [D4.1] which sets the scope of the work, analysing the requirements of particular applications and deriving the associated research challenges, and [D4.2] which provides the specifications of various network information consolidation and content distribution techniques addressing these challenges. This deliverable documents the delta from [D4.2], including the complete version of specifications that were suppressed from the public version of [D4.2] and the final specifications of the distributed data management infrastructure and the interactive video distribution tree optimisation algorithm.

The distributed data management infrastructure addresses one of the challenges of any dynamic and large scale overlay network, which is maintaining an accurate view of its resources and their current engagement in the distribution of a large number of content items. Such a data management infrastructure needs to scale with the size of the overlay and be responsive to a large number of queries from all the overlay distributed functions. To address this problem, an n-casting system is proposed in section 3, whereby a querying overlay node can discover its closest, in terms of network proximity, n distinct resources performing a particular overlay function. N-casting builds a distributed indexing protocol on top of an hierarchical clustering of the network endpoints and uses a statistical data structure for the efficient compression of the routing information.

For a tree-based streaming system developed in the context of interactive video applications, the limited capacity provided by the user end systems (peers) may result in the construction of a delivery tree violating the latency upper bound requirement associated with interactive video. This holds true even for small-scale events and availability of resources within the overlay itself that would otherwise be sufficiently high to support all participants. One way to mitigate the limitations of a particular set of overlay resources is to introduce nodes in the overlay tree with higher fan-out degrees using the High Capacity Node network service provided by the network operators as described in [D3.3]. A Tree Optimisation algorithm designed to provide a structured approach to this problem is described in section 4.1. In the target scenario, participants join the overlay and form the end-system distribution tree. Once the latency threshold has been violated, the algorithm is triggered to optimise the distribution tree in a cost-effective manner by including of High Capacity Nodes in strategic locations.

Finally, this deliverable includes specifications for Consolidated Overlay View Functions and Caching Optimisation based on Social Network Data that were suppressed from the public version of [D4.2]. In section 2.1 we propose a model for studying the tradeoff between ISP costs and overlay optimality, which emanates in these cases where the overlay application needs to include additional considerations when determining its overlay topology and traffic matrix, including for example the upload capacity of the overlay nodes. Section 2.2 explores techniques based on voting systems for consolidating the subjective views of different network operators on the desirability of a particular overlay connection to a single value that can be used by the overlay. Section 4.2 presents a caching optimisation technique for long-tailed static content i.e. content that is popular among small groups of users like the majority of the content in online social networks, taking into account ISP preferences for transmitting traffic during off-peak hours.

This deliverable concludes the design and specifications work in WP4. The techniques specified here and in previous deliverables will be evaluated in WP6, and the final evaluation results will be published in D6.2 at the end of the year.

2. CONSOLIDATED OVERLAY VIEW

The following sections elaborate on two different aspects of consolidating information received through CINA. First, we are proposing a model for the tradeoff in consolidating ISP costs with the benefits of the overlay. This work exploits some general properties underlying the function of the cost of an ISP and the benefit of an overlay with respect to traffic volume and it can be also found in [LMC+12]. Finally, we are presenting an approach on consolidating costs that represent the preferences of different ISPs using range voting and random ballots techniques (see also [LMG+12] for the published paper).

2.1 Overlay-ISP Cooperation Tradeoff

2.1.1 Problem Statement

The increasing demand for efficient content distribution using the Internet has fuelled the deployment of varied techniques such as peer-to-peer overlays, content distribution networks and distributed caching systems. These have had considerable impact on ISP infrastructure demand, motivating the development of protocols that enable mutually beneficial cooperative outcomes between overlays and ISPs. We propose a parameterised *cooperation utility* that can be used to study the tradeoff between the benefit that an overlay obtains from the ISPs that carry its traffic and the costs that it imposes on them. Using this utility, we find a closed-form expression for the optimal resource allocation given a particular cooperation tradeoff, subject to both minimal benefit and maximal cost constraints. The properties of this model are then explored through simulations in both a simple illustrative scenario and a more complete one based on network measurements and commonly used resource allocation policies. Since this model is based only on basic assumptions regarding overlay and ISP preferences, it is implementation-independent and can be used to explore the common foundations of a large class of ISP-aware overlays. Further, since the solution is analytic, it has very modest computational demands and can be used in large-scale simulations.

2.1.2 Approach

User demand for content distributed over the Internet has increased enormously in the last decade. As a result, diverse solutions based on network overlays have been deployed to make content distribution faster and more scalable. These include peer-to-peer systems, content distribution networks and distributed caching infrastructures, and we shall group them under the name of content distribution overlays (CDOs). In this section we present a model that can be used to describe a range of cooperative behaviours between CDOs and their underlying ISPs, taking into account the preferences of both.

If one considers the traffic matrix of greatest benefit to a given CDO, it is clear that it will depend on its preferences regarding cost, QoS, resource availability, replication and data caching. On the other hand, if one considers the traffic matrix of greatest benefit to an ISP providing the overlay with network connectivity, it will depend on its infrastructure and transmission costs, the background traffic that it carries and its traffic engineering policies. Although tensions may therefore arise between the preferences of the overlay and those of the ISP, the existence of mutually beneficial outcomes arising from ISP-CDO cooperation has been extensively documented [XYK+08, AAF08, BLD10, SCPR09, AFS07, DLL+11, RLY+11, BCC+06, CB08, PFA+10, JZSRC08, KMK+09]. This has sparked interest not only within the network research community, but also within standardisation working groups [PMG09]. Usually, these works investigate particular tradeoffs between overlay optimality and ISP costs in the context of specific protocols or applications. A more general cost-benefit model for these tradeoffs, developed from basic assumptions describing the preferences of both ISPs and CDOs, can be a useful tool in the understanding of the common foundations that they share. A study on how to express the preferences of the ISP determined based on their transit traffic costs can also be found in [D3.2], section 4.5.

The main contributions of this section are a parameterised cooperation utility that can be used to describe the cost-benefit tradeoffs of ISP-aware content distribution overlays, and a closed-form solution for the optimal tradeoff that arises from it. Our model starts from a set of basic assumptions regarding both the benefits that overlays can obtain and the costs that they impose on the ISPs that carry their traffic, and goes on to provide benefit and cost functions that satisfy them. A utility function is then presented that can be used to describe the tradeoff between CDO benefit and ISP cost. This utility then becomes the objective in a constrained maximisation problem, which is solved to provide a closed form solution for the CDO traffic matrix that embodies the optimal tradeoff. This analytic solution greatly reduces the computational effort involved in performing simulations using our model, allowing it to scale easily to overlays with several million links using off-the-shelf computational resources. Thus, our model can facilitate the investigation of the traffic dynamics of large ISP-aware CDOs.

2.1.2.1 Model Components

We commence our analysis by defining the central components of our model, as shown in Figure 1.

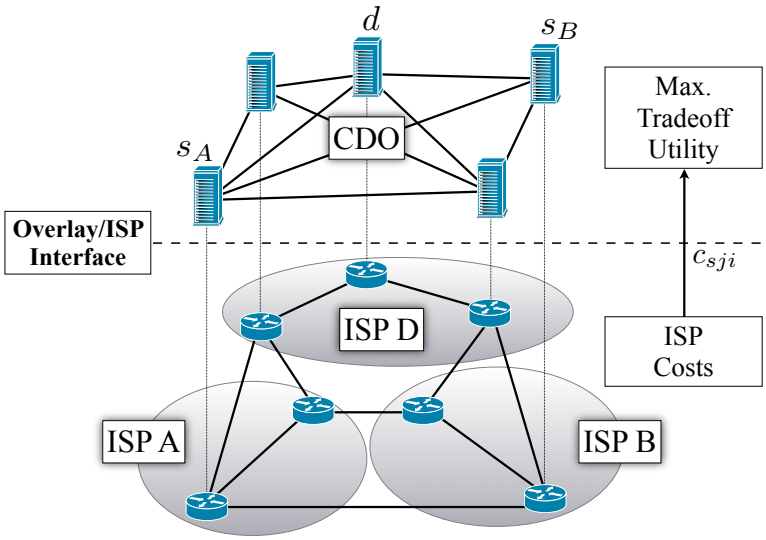

Figure 1: The Overlay-ISP Boundary between a CDO and its underlying ISPs.

- *Content Distribution Overlays (CDO)*, overlay networks formed by a set of nodes placed across the Internet and providing content retrieval services to end users. Examples would be peer-to-peer networks, managed overlay networks for the distribution of multimedia streams or content delivery networks (CDNs).
- *Internet Service Providers (ISP)*, which provide connectivity services to end customers and CDOs.

The benefit that a CDO obtains from its underlying ISPs increases when it delivers more content to its users, and when it does so in a more timely and reliable manner. To this end, the CDO needs to inject more traffic into the network, and to delegate greater system responsibilities to those nodes that provide better service to the end users in terms of network performance metrics such as delay and loss. However, the particular metrics that characterise the desirability of a node depend on the application. For interactive applications like gaming, high latency is undesirable; for streaming content, high jitter increases buffering time, and for time-sensitive encoders where retransmissions are not an option, high loss may result in disruptions of the decoding process. In our model, we will use the end-to-end *flow quality* as a measure of the benefit enjoyed by the CDO from traffic between two nodes. In addition to a constant-quality case, we will consider a variable-quality case that captures access link congestion effects. With regards to ISP cost, we will use an end-to-end *flow cost* provided by the ISP through which the traffic enters the network. Although this cost aims to capture the preferences of ISPs regarding traffic profitability, it may not necessarily translate directly to the

monetary cost of buying new equipment or leasing additional capacity because it may also include intangibles such as increased congestion delays or extra management costs associated with higher link utilisation. As with flow quality, we will also consider a variable-cost case in which cost is expressed as a function of the amount of traffic carried over a particular CDO flow.

2.1.2.2 The Cooperation Utility

We model the tradeoffs in CDO-ISP cooperation by proposing a utility function that balances the benefit that the overlay gets from the services provided by its underlying ISPs and the costs that it imposes on them. For a given set of CDO and ISP preferences, including a desired cost-benefit tradeoff, this utility function can be used to assess the performance of different CDO traffic allocation policies when compared with the optimum. Alternatively, when the desired cost-benefit tradeoff is not known a-priori, but a lower bound on CDO benefit and upper bound on ISP cost are given, this utility can be used to provide the range of tradeoffs between CDO and ISP preferences in which cooperation is feasible.

Our model is based on the assumption that the parameters that describe the ISP and CDO preferences are known. In addition, we will assume that the CDO obtains overlay link qualities itself, and that it is informed of overlay link costs from its underlying ISPs. If, however, the problem of interest is the modelling and analysis of existing ISP-aware CDO protocols, the parameters representing these preferences can be fitted from CDO-ISP interaction data.

We now formalise the cooperation utility optimisation problem. For each node in the CDO we consider a utility function U_i that combines the benefits that it can obtain given a particular traffic matrix with the costs that such a traffic matrix will impose on its underlying ISPs. We then maximise this utility, taking as input the relevant flow costs and qualities. This will yield the optimal CDO traffic matrix in terms of a set of bandwidth allocations to traffic flows. We define a flow as a 3-tuple *(s,j,i)* consisting of an ISP *s*, an origin node *j* and a destination node *i*. Conceptually, a flow is a representation for the traffic flowing from node *j* to node *i* through an ISP s available to *j*. Each flow will be annotated with a flow volume b_{sji} which represents the amount of traffic that the flow carries, a flow cost per unit bandwidth c_{sji} provided by *s*, and a flow quality q_{sji} estimated or measured by the overlay.

We propose that each one of the nodes of the overlay will have a utility (see Table 1 for variable definitions) such that

$$U_i = \alpha_i B_i - \varepsilon_i C_i,$$

in which the benefit and cost terms are

$$B_i = \left(\sum_{s \in I, j \in N} b_{sji}^{\beta_i} q_{sji}^{\gamma_i} \right)^{\delta_i}, \; C_i = \left(\sum_{s \in I, j \in N} b_{sji}^{\zeta_i} c_{sji}^{\eta_i} \right)^{\theta_i}$$

and where α_i, β_i, γ_i, δ_i, ε_i, ζ_i, η_i and θ_i are cost-benefit parameters that can be tuned to capture the preferences of both the CDO and its underlying ISPs. The first term in U_i models the benefit that node *i* obtains from the aggregate traffic that it receives from all other nodes; the second term, the aggregate cost that the ISPs are exposed to by carrying this traffic. Rather than relying in intricate protocol specifications or detailed ISP business models, we aim to find a clean, general model based on assumptions that are as basic as possible. For B_i, this led us to select the proposed functional form because it captures several intuitions about CDO preferences. These are discussed below.

$U_i \in R_{\geq 0}$	Utility that the CDO obtains from traffic flows terminating on *i*
$b_{sji} \in R_{\geq 0}$	Amount of traffic flow from node *j* to node *i* entering the network through ISP *s* (bandwidth)

$q_{sji} \in R_{\geq 0}$	Benefit that a unit bandwidth flow from node *j* to node *i* over ISP *s* provides to the CDO (quality)
$c_{sji} \in R_{\geq 0}$	Cost per unit bandwidth that ISP *s* announces to flows from *j* to *i* entering the network through *s* (cost)
$\mathcal{I}$	Set of all ISPs
N	Set of all CDO nodes
$\alpha_i \in R_{\geq 0}$ $\varepsilon_i \in R_{\geq 0}$	Tunable parameters that describe the relative importance of CDO cost and ISP benefit
$\beta_i \in (0,1)$ $\gamma_i \in (0,1]$ $\delta_i \in (0,1]$	Tunable parameters that describe diminishing returns in CDO benefit. For the analysis of specific protocols, these can be obtained from conventional least squares fitting.
$\zeta_i \in (0,1)$ $\eta_i \in (0,1]$ $\theta_i \in (0,1]$	Tunable parameters that describe diminishing returns in ISP cost. For the analysis of specific protocols, these can be obtained from conventional least squares fitting.

Table 1: Overlay-ISP Cooperation Tradeoff Variable Definitions

- *Increasing benefit with increasing flow volume* ($\alpha_i > 0,\ \beta_i > 0,\ \delta_i > 0$). In a capacity constrained scenario, the best nodes would only be able to provide service to a subset of end users, forcing the rest to rely on less desirable nodes and leading to reduced CDO benefit. Since increased flow volume ameliorates this, it results in an increased benefit for the overlay.
- *Increasing benefit with increasing quality* ($\gamma_i > 0,\ \delta_i > 0$). We assume that overlay links will be annotated with a *quality* q_{sji}, so that transferring the same amount of traffic between two nodes yields greater benefit if the quality of the overlay link between them increases. This effect is particularly important for CDOs carrying real-time traffic such as interactive media and gaming.
- *Diminishing marginal benefit on increasing flow volume* ($\beta_i < 1$). This models the fact that not all data available in a given node is equally useful. Thus, any given node will experience decreasing marginal benefit from increasing amounts of received traffic from another given node.
- *Non-increasing marginal benefit on increasing flow quality* ($\gamma_i \leq 1$). In many cases, such as voice or video streaming, once the quality of the received stream is high enough to decode the stream in time, no further improvement will be achieved by increasing the quality of overlay flows. Thus, benefit increases with quality, but only with with diminishing returns.
- *Non-increasing marginal benefit on the number of incoming flows* ($\delta_i \leq 1$). We assume that different nodes might have access to different kinds of content of interest to a particular node, so that benefit increases with the number of sender nodes that a given node has. However, it is improbable that all nodes will yield equivalent usefulness to the receiving node. Consequently, the benefit from connecting to an increasing numbers of nodes will increase at a decreasing rate.

Conversely, the cost function C_i captures several intuitions regarding the ISP preferences, which are now presented.

- *Increasing cost with increasing flow volume* ($\varepsilon_i > 0,\ \zeta_i > 0,\ \delta_i > 0$). We assume that, for a fixed cost-per-bit, the delivery of increasing amounts of traffic between overlay nodes imposes an increasing cost on ISPs. This is not only due to an increased contribution to direct 95th percentile billing, but also due to potentially increased congestion effects and other variable costs.

- *Increasing cost with increasing infrastructure cost* ($\eta_i > 0,\ \theta_i > 0$). We assume that transferring the same amount of traffic between two nodes imposes greater costs if the cost of the underlying network infrastructure is higher. This property can be used to model the cost characteristics of different network bearer services, such as PDH/SDH or GbE, or situations where an ISP offers tiered services with differentiated costs.
- *Non-increasing marginal flow volume cost* ($\zeta_i \leq 1$). This models the fact that Internet connectivity to a particular host imposes fixed costs, usually related to the provision of physical layer infrastructure. Thus, cost increases disproportionally for the first units of provisioned capacity.
- *Non-increasing marginal infrastructure cost* ($\eta_i \leq 1$). This allows the modelling of economies of scale in traffic aggregation, which lead to reduced costs-per-bit.
- *Non-increasing marginal cost for increasing number of nodes communicating with an overlay node* ($\theta_i \leq 1$). This allows the modelling of economies of scale in colocation and port density. Once a node has been provided with resources, providing resources to other nearby nodes can be done at a reduced cost per node.

2.1.3 Specifications

In our model, the *ISP-CDO Cooperation Problem* is solved by maximising the CDO cooperation utility taking the preferences of the CDO and ISPs as given by the cost-benefit tradeoff parameters α_i, β_i, γ_i, δ_i, ε_i, ζ_i, η_i and θ_i $\forall i \in N$. We formulate the problem as

$$\underset{b_{sji} \in R_{\geq 0}}{Maximise}: \quad U = \sum_{i \in N} U_i = \sum_{i \in N} \alpha_i B_i - \varepsilon_i C_i .$$

For this section, we will consider two cases in turn. In the first one the CDO is assumed to have no minimum requirements. This assumption will be dropped for the second, more general case where we consider both a minimum benefit that the CDO requires from its underlying ISPs and a maximum cost that it is willing to impose on them.

2.1.3.1 Case 1: No Operational Constraints

First, we will assume that the CDO has no requirements regarding either the benefits that it obtains or the costs it imposes on its underlying ISPs. Thus, we concentrate our attention on finding the optimal b_{sji} for given q_{sji}, c_{sji}, and additional model parameters. Since only non-restricted optimisation is required, we can apply first order conditions directly to the problem above. This leads to the following system of equations

$$\frac{\partial U}{\partial b_{sji}} = \sum_{i \in N} \alpha_i \frac{\partial B_i}{\partial b_{sji}} - \sum_{i \in N} \varepsilon_i \frac{\partial C_i}{\partial b_{sji}} = 0,$$

where B_i and C_i are given as above. For clarity reasons, for now we will disregard congestion and economy of scale effects, thus making q_{sji} and c_{sji} constant (we will re-introduce the notion of costs and qualities as functions of b_{sji} later).

Under these assumptions, the optimisation problem is separable and the first order conditions become

$$\alpha_i \frac{\partial B_i}{\partial b_{sji}} - \varepsilon_i \frac{\partial C_i}{\partial b_{sji}} = 0,$$

where the marginal benefit and cost terms are

$$\frac{\partial B_i}{\partial b_{sji}} = \delta_i \frac{\beta_i b_{sji}^{\beta_i - 1} q_{sji}^{\gamma_i}}{\left(\sum_{s \in I, j \in N} b_{sji}^{\beta_i} q_{sji}^{\gamma_i} \right)^{1-\delta_i}},$$

$$\frac{\partial C_i}{\partial b_{sji}} = \theta_i \frac{\zeta_i b_{sji}^{\zeta_i - 1} c_{sji}^{\eta_i}}{\left(\sum_{s \in I, j \in N} b_{sji}^{\zeta_i} c_{sji}^{\eta_i} \right)^{1-\theta_i}},$$

for each overlay node *i*. It is clear that the denominators in both expressions are independent from both *j*, the origin of the flow, and from its ingress ISP *s*; they are only a function of b_{sji}, the desired traffic matrix, and *i*, the node that is receiving the flow and hence assessing its utility. Thus, if we express both in terms of another, arbitrary flow terminating on the same node *i* but originating on a different origin node *m* and entering the network through a different ISP *n*, it can be shown that

$$\left(\frac{b_{sji}}{b_{nmi}} \right)^{\zeta_i - \beta_i} = \frac{q_{sji}^{\gamma_i}}{c_{sji}^{\eta_i}} \frac{c_{nmi}^{\eta_i}}{q_{nmi}^{\gamma_i}}.$$

This means that, discounted by a diminishing returns exponent $\zeta_i - \beta_i$, the ratio between the bandwidth allocated to two flows (*s*,*j*,*i*) and (*n*,*m*,*i*) terminating in the same node *i* will be equal to the ratio between their **cost-benefits**, defined as the ratios between their qualities and their costs. In particular, if we define the *preference-modified cost-benefit* μ_{sji} as

$$\mu_{sji} = \left(\frac{q_{sji}^{\gamma_i}}{c_{sji}^{\eta_i}} \right)^{\frac{1}{\zeta_i - \beta_i}},$$

we see that the first order conditions above imply that

$$\frac{b_{sji}}{b_{nmi}} = \frac{\mu_{sji}}{\mu_{nmi}}.$$

Thus, we see that the solution to the unconstrained optimisation problem will provide overlay traffic allocations b_{sji} proportional to the μ_{sji} associated with (*s*,*j*,*i*). Using the previous definitions, the first order conditions can be solved in the standard manner. This solution is cumbersome but straightforward, and is omitted for brevity. For the unconstrained case, we find that b_{sji}, the optimal bandwidth allocation for a flow between nodes *j* and *i* using ISP *s* as an ingress, can be expressed as

$$b_{sji} = \left(\frac{\alpha_i}{\varepsilon_i} \psi_i \right)^{\xi_i} \mu_{sji}$$

where

$$\psi_i = \frac{\beta_i \delta_i}{\zeta_i \theta_i} \frac{\left(\sum_{s \in I, j \in N} \mu_{sji}^{\zeta_i} c_{sji}^{\eta_i} \right)^{1-\theta_i}}{\left(\sum_{s \in I, j \in N} \mu_{sji}^{\beta_i} q_{sji}^{\gamma_i} \right)^{1-\delta_i}} \quad \text{and} \quad \xi_i = \frac{1}{\zeta_i \theta_i - \beta_i \delta_i}.$$

The set of flow volumes defined by this solution represent an optimal tradeoff between the CDO qualities q_{sji} and the costs c_{sji} announced by ISPs, given their respective preference parameters.

2.1.3.2 Case 2: Binding Budget Constraint for the Overlay

We now address the case where the CDO has operational constraints. To this end, we propose an improved model which, as we shall see, is a simple extension of that of the previous section. This new optimisation problem can be stated as

$$\underset{b_{sji}\in R_{\geq 0}}{Maximise}: \quad U=\sum_{i\in N}U_i=\sum_{i\in N}\alpha_i B_i-\varepsilon_i C_i$$

$$Subject\ to: \quad \sum_{i\in N}\left(\sum_{j\in N,s\in I}b_{sji}^{\beta_i}q_{sji}^{\gamma_i}\right)^{\delta_i}=\sum_{i\in N}B_i\geq B_{\min}$$

$$\sum_{i\in N}\left(\sum_{j\in N,s\in I}b_{sji}^{\zeta_i}c_{sji}^{\eta_i}\right)^{\theta_i}=\sum_{i\in N}C_i\geq C_{\max}$$

where B_{min} is the minimum benefit tolerable to the overlay, and C_{max} is the maximum aggregate cost that the overlay is willing to impose on all the ISPs that provide it with connectivity services. The solution to this problem is a simple extension to the unconstrained solution, with a slightly expanded Lagrangean that leads to the first order optimality conditions

$$(\alpha_i+\lambda_B)\frac{\partial B_i}{\partial b_{sji}}-(\varepsilon_i+\lambda_C)\frac{\partial C_i}{\partial b_{sji}}=0,$$

along with the two *complementary slackness* conditions

$$\lambda_B\left(B_{\min}-\sum_{i\in N}B_i\right)=0,$$

$$\lambda_C\left(\sum_{i\in N}C_i-C_{\max}\right)=0.$$

In the previous expressions, λ_B corresponds to the Lagrange multiplier associated with overlay benefit and λ_C corresponds to the Lagrange multiplier associated with ISP costs.

We seek an expression for b_{sji}^*, the solution to the budget-constrained problem. The derivation proceeds as in the previous case, and we have that

$$b_{sji}^*=\left(\frac{\alpha_i+\lambda_B}{\varepsilon_i+\lambda_C}\psi_i\right)^{\xi_i}\mu_{sji}=\left(\frac{1+\frac{\lambda_B}{\alpha_i}}{1+\frac{\lambda_C}{\varepsilon_i}}\right)^{\xi_i}b_{sji}$$

where b_{sji} is the solution of the unconstrained problem and represents the flow volume that would have been allocated to a flow from node *j* to node *i* entering the network through ISP *s*, *had no constraints been active*. In this case, $\lambda_C=\lambda_B=0$ and b_{sji}^* reduces to b_{sji}. For the simulation, we find λ_B, λ_C and b_{sji}^* using standard dual decomposition techniques [BV09]. Thanks to this analytic solution, and by allocating an independent thread to each peer in the optimisation solver, it is possible to take full advantage from the superior performance of multicore architectures.

2.1.3.3 Congestion and Economies of Scale

Having solved the ISP-CDO cooperation problem for constant q_{sji} and c_{sji}, we now expand our scope to consider $\tilde{q}_{sji}$ and $\tilde{c}_{sji}$, equivalent expressions that are functions of b_{sji}. To keep the model compatible with the solutions that we have already found, we will make two assumptions. The first one is that $\tilde{q}_{sji}$ and $\tilde{c}_{sji}$ are functions with *constant elasticity*; the second one is that their elasticities are *functions of i only*. When taken together, and if we restrict our attention to the Cobb-Douglas class of functions [CD28] widely used in microeconomics, this means that we will consider cost and quality functions of the form

$$\tilde{q}_{sji} = q_{sji} b_{sji}^{E_i^q},$$
$$\tilde{c}_{sji} = c_{sji} b_{sji}^{E_i^c}$$

where E_i^q is the *self-congestion* elasticity, E_i^c is the *economy-of-scale* elasticity, and q_{sji} and c_{sji} correspond to the constant quality and cost introduced earlier. The reason behind the naming of E_i^q and E_i^c can be found in the standard definition of elasticity [GR04], and is indicative of their function in the model. In particular, the magnitude of E_i^q will represent the percent decrease in overlay link quality q_{sji} with a percent increase in b_{sji}, and the magnitude of E_i^c will represent the percent decrease of per-unit-bandwidth cost c_{sji} with a percent increase in b_{sji}. Therefore, E_i^q will be used to model the effect of traffic volume on flow quality, and will be a measure of congestion in the access link of node *i*; conversely, E_i^c will be used to model the effect of traffic volume on flow cost, and will be a measure of economies of scale in the access link of node *i*. Since quality will be reduced with increased traffic flow, $E_i^q < 0$, and since economy of scale effects reduce the cost per bit, $E_i^c < 0$ as well.

The rationale behind this model of access link congestion and economies of scale is that it allows us to consider the effect of $\tilde{q}_{sji}$ and $\tilde{c}_{sji}$ as an additive constant. Consider the effect of replacing q_{sji} and c_{sji} in U_i with $\tilde{q}_{sji}$ and $\tilde{c}_{sji}$: it amounts to using modified $\tilde{\beta}_i$ and $\tilde{\xi}_i$ so that

$$\tilde{\beta}_i = \beta_i + \gamma_i E_i^q,$$
$$\tilde{\xi}_i = \zeta_i + \eta_i E_i^c.$$

To see how substituting $\tilde{\beta}_i$ and $\tilde{\xi}_i$ into the previous expressions changes our model, we note that b_{sji} will increase with increasing $\tilde{\beta}_i$, and it will decrease with increasing $\tilde{\xi}_i$. As $\left|E_i^q\right|$ increases, $\tilde{\beta}_i$ will decrease and b_{sji} will decrease as well; as $\left|E_i^c\right|$ increases, $\tilde{\xi}_i$ will decrease, and b_{sji} will increase. Thus, a propensity for congestion in the access link of *i* can be modelled with a large $\left|E_i^q\right|$, and the overlay will react with a general reduction in traffic towards *i* as flow volume increases. Conversely, if $\left|E_i^c\right|$ is large, denoting good cost efficiency, the overlay will react by increasing uploads towards *i*.

2.1.3.4 Model Parameters

In order for the previously found solutions to remain a valid solution of the optimisation problem, we must impose limitations on some of the model parameters. We now explore these, along with their implications.

- *Positive traffic attraction* ($\tilde{\beta}_i > 0$). When $E_i^q = \frac{\beta_i}{\gamma_i}$, $\tilde{\beta}_i = 0$, making b_{sji} equal to zero for all *s* and *j*. Conceptually, this means that the overlay has found the access link of node *i* to be excessively

congested, and thus removes it from consideration. Further increases in the magnitude of E_i^q are ignored; traffic will resume when E_i^q decreases.

- *Positive traffic avoidance* ($\tilde{\xi}_i > 0$). As $E_i^c \to \frac{\xi_i}{\eta_i}$, $\tilde{\xi}_i \to 0$, and $b_{sji} \to \infty$. This happens when the cost reduction associated with increasing volume on flows towards destination *i* is so large that the overlay attempts to increase the volume of this flow as much as possible. For the remainder, we will assume that $\tilde{\xi}_i > 0$, and thus, that b_{sji} is finite.
- *Concave Utility* ($\tilde{\xi}_i - \tilde{\beta}_i \to 0$ and $\tilde{\xi}_i\theta_i - \tilde{\beta}_i\delta_i \to 0$). These two conditions are related with the existence of a well-defined maximum for U_i over an unrestricted domain. In both cases, violation of these conditions implies that costs grow more slowly than benefits, and it is thus optimal for the overlay to increase its traffic flows towards *i* as much as possible. For the remainder, we assume that both these conditions hold and that, therefore, a well-defined maximum exists.

2.1.3.4.1 Parameter Fitting

The basic model parameters can be tuned to describe the tradeoffs made by particular overlay protocols. For the unconstrained model, this can be achieved simply by applying a simple least squares fitting to the solution presented above. This will take a set of *N* measurement samples U_i^t, b_{sji}^t, c_{sji}^t and q_{sji}^t, $t \in [1 \ldots n]$, and produce estimations for the modelling parameters. A detailed analysis of parameter fitting is omitted for brevity.

2.1.3.5 Related Work

The expression of explicit interactions between overlays and ISPs is receiving increased attention by the research community (see, for instance, [XYK+08, AAF08, BLD10, SCPR09, AFS07, DLL+11, RLY+11, BCC+06, CB08, PFA+10, JZSRC08, KMK+09]). These studies present particular ways in which CDO construction optimality and ISP cost can be balanced, thus providing specific examples along the spectrum of tradeoffs considered by our model. Some of these works are now described.

One of the first works to focus in BitTorrent locality was [BCC+06], which relies on the ISP tagging each peer as either *local* or *external*. This allows the overlay to set a soft limit on *external* peers, biasing peer selection and giving preference to intradomain connections. In [AFS07], the authors present a system in which an *oracle* performs peer ranking according to the preferences of the ISP. Although many possible metrics are presented as candidates (AS path length, IGP metric distance, geographical information, expected delay or bandwidth and link congestion) the authors focus in AS path length. In [AAF08], this system is extended to take into account peer upload bandwidth, and in [PFA+10] it is further improved by allowing the oracle to enrich DNS responses with ISP-provided information. There has been particular attention to BitTorrent locality in the context of overlay-ISP interaction. In [SCPR09], the authors present a modified BitTorrent tracker that answers client queries for swarm members with peers that are either selected from the 25% closest peers as determined from delay synthetic coordinates, or randomly selected over the entire swarm. In [BLD10], modified BitTorrent trackers keep logs of the number of peers external to a given ISP that have been given to peers in that ISP, and keep this number under a given threshold. In addition to work considering the explicit communication of preferences between overlays and ISPs, there has been work on *implicit* optimisation of overlays according to ISP preferences. One such work is [CB08], a technique developed in the context of CDNs that relies on a client-only clustering of peers using their DNS resolutions for CDN content as a similarity metric.

One of the main contributions in this area is P4P [XYK+08], which became the basis for the main standardisation effort in the area [PMG09]. In its original presentation, [XYK+08] relied on the ISP aggregating peers into groups (PIDs) and providing a set of end-to-end prices between them. In that

case, the overlay was assumed to solve a particular minimisation problem arising from the dual decomposition of the ISP optimisation problem. The present work provides an alternative, overlay-centred view which remains compatible with [PMG09] while being easily re-parameterisable to reflect diverse ISP and CDO preferences.

Although there is ample evidence of the existence of alignment of incentives between overlays and ISPs, there is also evidence that points to incentive conflicts that require resolution. As an example, the user incentives for BitTorrent localisation have been called in to question. In [PMJ+09], the authors consider many of the popular localisation techniques described, and show that they can be frequently exploited by the ISP to increase its revenue, which can lead to average path length increases of up to 72.6%. Furthermore, if peer throughput is not limited by latency, BitTorrent localisation will fail to deliver good performance to the users. This points to an specific opportunity for a tradeoff-aware protocol, which could be analysed using the model presented.

2.2 ISP Preference Consolidation

2.2.1 Problem Statement

Overlays collaborate with ISPs by driving their topology formation processes using information that each node obtains from its local ISP using open interfaces (i.e. CINA, ALTO [PMG09] and P4P [XYK+08]). This interaction usually involves ranked or annotated lists of network regions called *PID*s, which constitute clusters of topologically equivalent overlay nodes. By improving overlay construction via biased node selection, these collaboration techniques can be beneficial in reducing interdomain traffic and increasing overlay performance [AAF08, BCC+06, BLD10, RLY+11].

Most studies to date have focused on using *network locality* as the basis for these annotated PID lists. However, CINA provides a good vehicle to implement other objectives such as reducing interdomain traffic costs or managing persistent traffic hotspots. These uses extend the study of Overlay/ISP collaboration into the realm of *asymmetric preferences*. Whereas locality-based costs are symmetric, e.g. have the same properties in both directions, costs based on other network metrics may not have this property. For instance, ISPs contractually agree their interdomain costs with their service providers, leading to cost asymmetries. An overlay link between nodes in ISPs A and B may be *cheap* to A but *expensive* to B, or vice-versa. Therefore, a traffic allocation which is desirable to a given ISP may be undesirable for another one, and achieving an appropriate tradeoff therefore requires information from both ISPs. Taking interdomain cost as an example, Figure 2 shows three multi-homed stub ISPs with arrows representative of the transit costs charged by their transport providers. Conventional operation drives A to bias overlay topology formation towards B, imposing upon it additional costs. Likewise, C could load balance equally between A and B, whereas biasing topology formation towards B and away from A would reduce the total cost.

2.2.2 Approach

We propose to address this problem through *consolidation of preferences*, a process whereby each CINA server provides a preference-annotated list of PIDs, which are collected by overlay peers and consolidated into a single preference-annotated list that represents an adequate tradeoff between the preferences of all ISPs involved. This list is then used by the overlay to drive its topology construction.

In the following section, the first contribution is the definition of a generic model describing ISP preference consolidation. We then present two preference consolidation strategies: *Shared Cost*, designed to provide a tradeoff for preference cost asymmetries, and *Low Cost*, designed to reduce the overall preference cost that the overlay imposes on all its underlying ISPs. In [D6.1] we provide details on the evaluation of the consolidation strategies with the objective to show that preference consolidation can provide ISPs with outcomes more aligned with their preferences than those provided by non-consolidated operation.

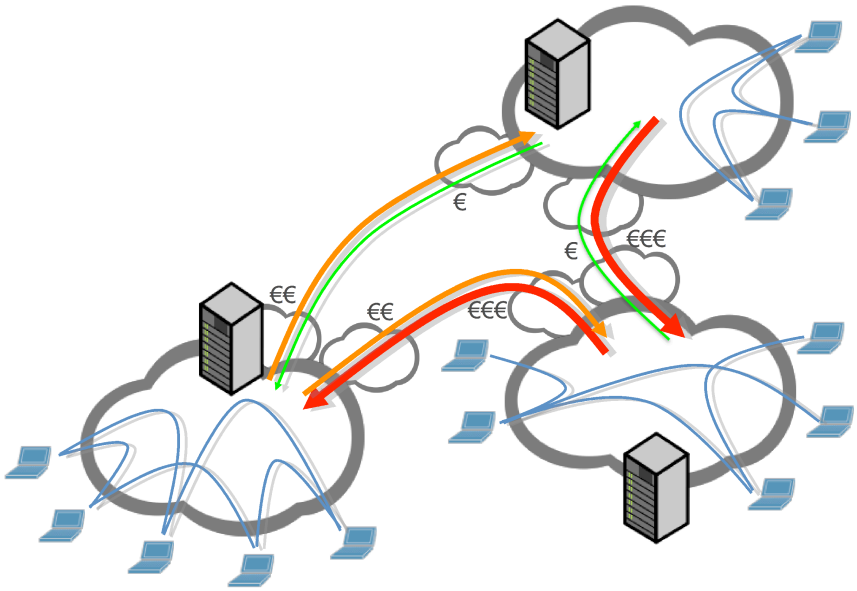

Figure 2: Asymmetric ISP preferences

2.2.3 Specifications

We consider an overlay with a presence in a set I of N_I ISPs, so that $N_I = |\mathsf{I}|$. We focus in the construction of the *service topology*, that is, the set of overlay links that nodes use to exchange large amounts of traffic (rather than much smaller flows of signalling information). We call nodes adjacent in the service topology as *neighbours*, and consider the formation of this topology on the basis of ISP-provided information. We will assume that once a node has selected another node as a neighbour, this will trigger a unidirectional traffic stream at a standard average rate. Node selection will be our only consideration for topology construction; overlay-specific algorithms related to incentives or content availability (e.g. *tit-for-tat*) will not be considered.

We will assume that each overlay node will require k neighbours, and that each ISP will provide an CINA server that influences the topology formation choices of each node. To this end, we assume that each ISP i decomposes the entire set of overlay nodes into clusters of nodes which are considered essentially indistinguishable from the point of view of overlay topology formation; these will be called PIDs. We will assume that each ISP i defines both a set $\mathsf{P}_I(i)$ of *internal* PIDs that correspond its own customers, and a set of PIDs $\mathsf{P}_E(i)$ that comprises all other overlay nodes. Since these two are disjoint, each ISP can then define a set $\mathsf{P}(i) = \mathsf{P}_I(i) \cup \mathsf{P}_E(i)$ that represents all PIDs, both internal and external (of course, $\mathsf{P}_I(i) \subset \mathsf{P}(i)$ and $\mathsf{P}_E(i) \subset \mathsf{P}(i)$). We define the creation of an *overlay link* between PID l and PID m as the topology formation decision by a node in l selecting a node in m as its neighbour in the overlay. Then, we assume that each ISP i defines a *preference cost function* $c_i(l,m) : \mathsf{P}_I(i) \times \mathsf{P}(i) \rightarrow \mathsf{R}_+$ that assigns preference costs to overlay links. Note that we assume that an ISP i will only define preference costs for overlay links initiating in PIDs local to i (our model does not include transit ISPs, as in other current interfaces these do not announce preferences). Hence, $c_i(l,m)$ models how strongly does ISP i express a preference for nodes from an internal PID l to select nodes in PID m as neighbours in the overlay (m can be any PID, including internal ones). This preference cost function forms the basis for ISP-influenced overlay topology construction.

2.2.3.1 Preference Consolidation

In this paper we consider an multiple-ISP case in which each ISP i generates an inter-PID cost function $c_i(l,m)$. To simplify the problem, we will assume that there is a well-known decomposition of overlay nodes into PIDs, and that all PIDs are considered internal only for a single ISP. This means that there is a universal P so that $\mathsf{P}(i)=\mathsf{P}$ for all i, and that internal PID sets $\mathsf{P}_I(i)$ are disjoint subsets of P. Hence, in $c_i(l,m)$, the ISP i is uniquely defined by the PID l, and we can directly use the shorthand $c(l,m)$ where l implicitly defines i through a mapping $i=\pi(l)$ where $\pi(l):\mathsf{P}\rightarrow\mathsf{I}$ is a function that maps each PID l to its provider ISP i. Formally, we define the *generic cost* function $c(l,m):\mathsf{P}\times\mathsf{P}\rightarrow\mathsf{R}_+$. This function maps any given pair of overlay PIDs l and m to a positive value representing the preference that ISP $\pi(l)$ has for nodes in PID l to create overlay links with nodes in m. To simplify matters further and aid comparison and aggregation, we assume that all ISPs agree on a common set of cost semantics.

Let the space of all generic cost functions $c(l,m)$ be called C. A *consolidation function* is a function $F:\mathsf{C}\rightarrow\mathsf{C}$ that can be used to generate a single preference cost function from the partial preference cost functions provided by participant ISPs. Formally, if we denote our consolidated preference cost function as $\kappa(l,m)$, it follows that $\kappa(l,m)=F(c(l,m))$, where F can be designed so that $\kappa(l,m)$ has specific properties.

Since our objective is to compare these preference functions quantitatively, we require a baseline that represents default behaviour and from which we can measure any potential improvements. For this, we will use a F equal to the trivial identity mapping, so that $\kappa(l,m)=c(l,m)$. We will call this the *default* consolidation. Note that this corresponds to the case in which every node queries their local CINA server and ignores information provided by CINA servers of other nodes.

2.2.3.2 Overlay Topology Construction

We now address the modelling of how a particular consolidated cost function impacts the overlay topology formation process. We assume that each node will query the overlay for a set of candidate neighbours, and then select randomly from this set (this is reminiscent of CINA interaction, with the functions of the CINA server being provided by the overlay itself). The overlay can bias topology construction by providing candidate nodes in given PIDs with non-uniform probabilities. From a modelling standpoint, we simulate topology formation by determining how many nodes (and from which PIDs) will select nodes from a given PID as neighbours.

We start by defining a *population* function $N(l):\mathsf{P}\rightarrow\mathsf{N}$ that assigns to each PID l the number of overlay nodes residing in it. In addition, we define a *topology construction* function $p(l,m):\mathsf{P}\times\mathsf{P}\rightarrow[0,1]$ that denotes the proportion of nodes in PID l that select nodes in PID m as their neighbours. Since $p(l,m)$ can also be interpreted as the probability of a node in l choosing a node in m as a neighbour, we have that $\sum_{m\in\mathsf{P}}p(l,m)=1$. We will relate $p(l,m)$ with $\kappa(l,m)$ through a *preference-topology* function G, so that $p(l,m)=G(\kappa(l,m))$. Hence, G will model the impact that a given set of consolidated preference costs $\kappa(l,m)$ will have on $p(l,m)$.

2.2.3.3 Consolidated Topology Construction

In the following we propose and evaluate two *consolidated topology construction* strategies: *Shared Cost* (**SC**) and *Low Cost* (**LC**). Both these strategies consist of a consolidation function F and a preference-topology function G. In addition, to serve as a baseline for comparison, we also describe a third algorithm that corresponds to a non-consolidated usage of the generic preference costs $c(l,m)$ provided by each CINA server.

2.2.3.3.1 Shared Cost (SC)

The objective of this consolidation strategy is to find a tradeoff for the cost asymmetries present in the generic preference cost function $c(l,m)$. To achieve this, we define F as function that makes $\kappa(l,m)$ symmetric by assigning it the average of $c(l,m)$ and $c(m,l)$ weighed by the populations of m and l:

$$\kappa(l,m) = \kappa(m,l) = \frac{N(l)c(l,m) + N(m)c(m,l)}{N(l) + N(m)}. \quad (1)$$

We define G by considering $p(l,m)$ as inversely proportional to $\kappa(l,m)$, so that

$$p(l,m) = \frac{\kappa(l,m)^{-1}}{\sum_{n \in P} \kappa(l,n)^{-1}}. \quad (2)$$

This means that, given a consolidated set of preference costs $\kappa(l,m)$, the overlay will select overlay links (l,m) with low $\kappa(l,m)$ with a higher frequency than those with high $\kappa(l,m)$.

2.2.3.3.2 Low Cost (LC)

The objective of this consolidation strategy is to induce an overlay topology that reduces the overall preference cost that the overlay imposes on all its underlying ISPs. To achieve this, the overlay will only create overlay links between PIDs l and m if $\kappa(l,m)$ is among the q lowest preference cost alternatives for PID l. More formally, we only allow nodes in a given PID l to become neighbours of nodes belonging to PIDs in a subset Q_l of cardinality q. We define Q_l to include the first q PIDs in a list of P sorted by the preference cost $c(l,m)$. Hence, the PIDs in Q_l will be the q PIDs that have the lowest cost from l, and we have that

$$\kappa(l,m) = \begin{cases} 1 & if\ l \in Q_l \\ \infty & otherwise. \end{cases} \quad (3)$$

We define G in the same way as for the **SC** case, so that (2) still holds. For the specific case of **LC**, then, we have that

$$p(l,m) = \begin{cases} \frac{1}{q} & if\ l \in Q_l \\ 0 & otherwise. \end{cases} \quad (4)$$

2.2.3.3.3 Default

The objective of this consolidation strategy is to model the default situation where each node queries its local CINA server and performs node selection on the basis of this information only. For this strategy, we define F as the identity function so that $\kappa(l,m) = c(l,m)$, and use G so that (2) holds. Hence, $p(l,m)$ is inversely proportional to $c(l,m)$.

2.2.3.4 Related Work

The expression of explicit interactions between overlays and ISPs is receiving increased attention by the research community (see, for instance, [AAF08, JZSRC08, KMK+09]). One of the first works to focus in BitTorrent locality was [BCC+06], which relies on the ISP tagging each node as either *local* or *external*. This allows the overlay to set a soft limit on *external* nodes, biasing node selection and giving preference to intradomain connections. In [AFS07], the authors present a system in which an *oracle* performs node ranking according to the preferences of the ISP. Although many possible

metrics are presented as candidates (IGP metric distance, geographical information, expected delay, etc.) the authors focus in AS path length. In [AAF08], this system is extended to take into account node upload bandwidth, and further improved upon by allowing the oracle to enrich DNS responses with ISP-provided information [PFA+10]. Another contribution in this topic is P4P [XYK+08], which became the basis for one of the main standardisation efforts in the area [AFS07]. In its original presentation, [XYK+08] relied on the ISP aggregating nodes into groups called PIDs and then providing a set of end-to-end costs between the PIDs. These ISP-provided prices are used by the overlay to calculate its traffic matrix.

Regarding the reduction of ISP costs, multiple solutions have been proposed. Examples include ISP peering , IP multicast, content distribution networks and P2P localisation [BLD10, SCPR09, CB08]. Another technique that has received increased attention from the research community is *traffic shaping*, that relies on reducing peak traffic volumes. This, in turn, reduces costs because traffic is usually billed on the basis of its 95-th percentile [DHKS09]. Particular approaches to reduce interdomain costs include simple rate-limiting/traffic shaping [MDGV11] or deferred transmission of delay-tolerant traffic [LSRS09]. Another proposal [SCG11] proposes multiple ISPs to cooperate by jointly purchasing bulk transit bandwidth. This can be used to save costs due to the economies-of-scale effect of subadditive pricing as well as burstable billing: not all ISPs transit their peak traffic during the same period. Finally, there has been some interest to approach the overlay-ISP interaction problem from the perspective of economics. As an example of this current of work, [F07] presents a *market-centric*, architectural view. Additional examples of the analysis of cooperative outcomes between ISPs and overlays include the use of the Shapley value [MCL+07].

3. DISTRIBUTED DATA MANAGEMENT INFRASTRUCTURE

3.1 Problem Statement

In ENVISION, a unique identifier is used for each type of overlay resource, e.g. content relaying nodes, content adaptation nodes, available high capacity nodes, etc., as well as for each content object that these resources are processing, storing or distributing, e.g. content relaying nodes relaying a particular video stream will use a dedicated identifier.

The resource discovery function should scale with the number of the information items that it needs to store and should be highly distributed, partitioning the resource information indexes and distributing the processing of resource discovery requests between the overlay nodes. This distributed processing then involves forwarding query messages between the overlay nodes and needs to be optimised for accuracy and speed.

The discovery of any type of resources involves two basic operations:

- the exact match of an identifier according to the desired type of resources and the particular content distribution overlay they are part of and
- the filtering and/or ranking of the matching resources based on application and network performance criteria, e.g. the smallest network delay to the querying node.

While the first part has been extensively studied in distributed environments, the second part and the combined problem are not widely addressed. This dual problem can be addressed using anycast and n-cast (also known as manycast) techniques. Anycast is used to route a message to *any* node that is a registered member of a specified group and is selected based on some proximity metric to the message originator node. Unlike anycast where the message is routed to the node which is determined to be the best fit for the selection criteria, in a n-cast system, the sender can specify *n* nodes where the message will be routed to. The *n* nodes that best meet the selection criteria will all receive the message. N-casting can be also positioned as an operation that fills the spectrum of network communication space between anycast and multicast [CYRK03].

The following sections provide a description of the system developed in ENVISION for discovering resources with n-casting, using the network delay between the overlay resource nodes as the member selection criteria.

In ENVISION, we have developed a system for discovering resources with n-casting, using the network delay between the overlay resource nodes as the member selection criteria.

The rest of this section has been suppressed from the public version of this deliverable.

4. OVERLAY RESOURCE OPTIMISATION AND CONTENT DISTRIBUTION

4.1 Interactive Video Content Distribution

4.1.1 Problem Statement

In this section we discuss a solution for a scenario where the application benefits from protocols and mechanisms developed within the ENVISION project. We focus on the design of a distributed content distribution system, which can be used by applications such as video conferencing and other small-to-medium scale live events broadcasting. For this class of applications low-latency delivery is of paramount importance.

Therefore the content distribution system needs to satisfy the following requirements (more details can be found in [D4.1]):

- The peer-to-peer system has to create an overlay that allows applications the distribution of media streams
- The overlay topology must be able to take information into account that minimise the end-to-end latency
- The system must be able to integrate network services into the overlay topology and optimise the topology accordingly

4.1.2 Approach

The Interactive Video Content Distribution (IVCD) system introduced in this section allows the system to create an overlay topology specifically for the distribution of interactive HD AV content across dynamic groups of users. Therefore routing and scheduling algorithms create a distribution topology which is optimised for the transmission of live media streams that are typical for an interactive conferencing application where a user group exchanges media data in real time. To achieve an immersive interactive user experience the focus of the content distribution algorithm is to minimise the end-to-end latency of the media streams between the users. Therefore the module that controls the overlay topology is able to implement different strategies that are able to demonstrate the effect of network information that can be gained by the CINA interface. We further study an optimisation procedure that enables the system to decide where and when to invoke CINA enabled network services, such as the High Capacity Node.

The implementation of the system and the various topology strategies implemented by the Tree Manager module have been described in full in deliverable [D4.2]. In this deliverable we discuss the final specifications of the Distribution Tree Optimisation procedure, which allows the overlay to decide where to integrate a service like the High-Capacity Node Service.

The rest of this section has been suppressed from the public version of this deliverable.

4.2 Caching Optimisation based on Social Network Data

4.2.1 Problem Statement

Online content distribution technologies have witnessed many advancements over the last decade, from large CDNs to P2P technologies, but most of these technologies are inadequate while handling unpopular or long-tailed[1] content. CDNs find it economically infeasible to deal with such content – the distribution costs for content that will be consumed by very few people globally is higher than

[1] in terms of views

the utility derived from delivering such content [ASKF10]. Unmanaged P2P systems suffer from peer/seeder shortage and meeting bandwidth and/or QoE constraints for such content.

The problem of delivering such content is further exacerbated by two recent trends: the increasing popularity of user-generated content (UGC), and the rise of online social networks (OSNs) as a distribution mechanism that has helped reinforce the long-tailed nature of content. For instance, Facebook hosts more images than all other popular photo hosting websites such as Flickr [urlb], and they now host and serve a large proportion of videos as well [Faca]. Content created and shared on social networks is predominantly long-tailed with a limited interest group, specially if one considers notions like Dunbar's number [Dun92]. The increasing adoption of smartphones, with advanced capabilities, will further drive this trend.

In order to deliver content and handle a diverse user-base [HWLR08, Lin], most large distributed systems are relying on geo-diversification, with storage in the network [TFK11, Facb, Twi]. One can push or prestage content to geo-diversified PoPs closest to the user, hence limiting the parts of the network affected by a request and improving QoE for the user in terms of reduced latency. However, it has been shown that transferring content between such PoPs can be expensive due to bandwidth costs [LSYR11, For]. For long-tailed content, the problem is more acute – one can *push* content to PoPs, only to have it not consumed, wasting bandwidth. Inversely one can resort to *pull*, and transfer content only upon request, but leading to increased latencies and potentially contributing to the peak load. Given the factors above, along with the inability of current technologies to handle such content [ASKF10, Ken] while keeping bandwidth costs low, it would appear that distributing long-tailed content is and will be a difficult endeavor.

4.2.2 Approach

We present a system called TailGate that can distribute long-tailed content while lowering bandwidth costs and improving QoE. The key to distribution is to know (i) *where* the content will likely be consumed, and (ii) *when*. If we know the answers, we can *push* content where-ever it is needed, at a time before it is needed, and such that bandwidth costs are minimized for the underlying ISPs operating under peak based pricing schemes like 95th percentile pricing [bur]. The overlay application responsible for caching and distributing the content to the end users communicates with the ISPs where the users and the caching nodes reside, and retrieves through CINA their preferences regarding shifting traffic at particular times of the day. TailGate is an optimisation technique that can be part of an OSN application operating its own service infrastructure, or of a CDN responsible for caching content on behalf of an OSN (see section 4.2.3.4 for details on the possible deployment scenarios).

Although here we will focus on the 95th percentile pricing scheme, it needs to be stressed that lowering the peak is beneficial also under flat rate schemes or even with owned links since network dimensioning in both cases depends on the peak. Recent proposals like NetSticher [LSYR11] have proposed systems to distribute content between geo-diversified centers, while minimizing bandwidth costs. TailGate augments such solutions by relying on a hitherto untapped resource – information readily available from OSNs.

More specifically, TailGate relies on the rich and ubiquitous information – friendship links, regularity of activity and information dissemination via the social network. TailGate is built around the following notions that dictate consumption patterns of users. First, users follow strong diurnal trends while accessing data [SFKW09]. Second, in a geo-diverse system, there exist time-zone differences between sites. Third, the social graph provides information on who will likely consume the content. At the center of TailGate is a scheduling mechanism that uses these notions. TailGate schedules content by exploiting time-zone differences that exist, and trying to spread and *flatten* out the traffic caused by moving content. The scheduling scheme enforces an informed *push* scheme (described in section 4.2.3), reduces peaks and hence the costs. In addition, the content is pushed to the relevant

sites before it is likely accessed – reducing the latency for the end-users. We designed TailGate to be be simple and adaptable to different deployment scenarios.

In order to first understand characteristics of users that can be used by Tailgate and if these characteristics are useful, we turn to a large dataset collected from an OSN (Twitter), consisting of over 8M users and over 100M content links shared. This data helps us understand where requests can come from as well as give us an idea of when.

For the sake of exposition, we describe a generic distributed architecture that will provide the template for the design and analysis of TailGate. In section 4.2.3.4, we show how this architecture can be used for different scenarios – OSN providers and CDNs. After describing the architecture, we provide a simple motivating example. At the end of the section we list the requirements that a system like TailGate needs to fulfill. TailGate is also documented in [OSL12].

4.2.2.1 Architecture

We consider an online service having users distributed across the world. In order to cater to these users, the service is operated on a geo-diverse system comprising multiple points-of-presence (PoPs) distributed globally. These PoPs are connected to each other by links. These links can be owned by the entity owning the PoPs (for instance, Google or a Telco-operated CDN) or the bandwidth on these links can be leased from network providers.

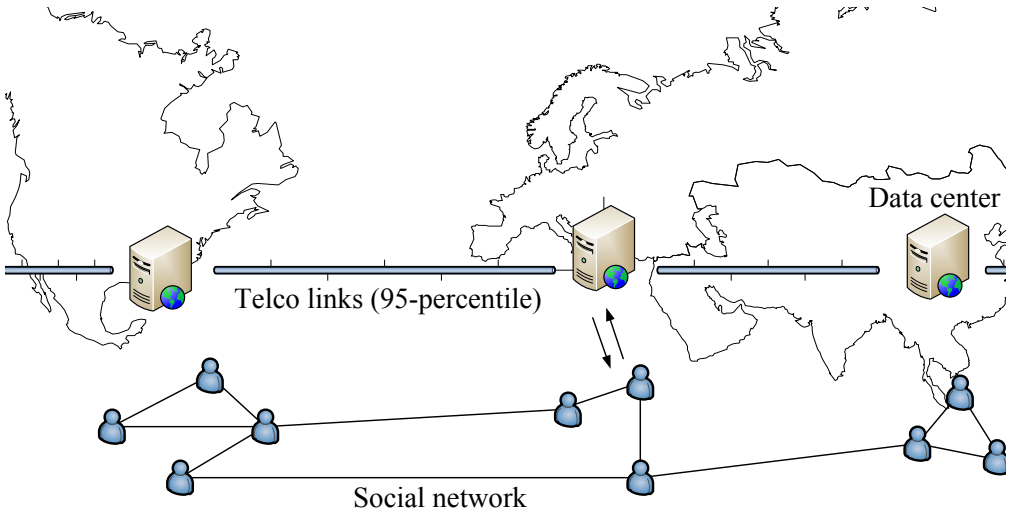

Figure 3: Generic distributed architecture: Servers or PoPs geo-distributed, handling content for geographically close users

Users are assigned and served out of their nearest (geographically) PoP, for all their requests. Placing data close to the users is a maxim followed by most CDNs, replicated services, as well as research proposals like Volley [ADJ+10]. Therefore all content uploaded by users is first uploaded to the nearest respective PoP. When content is requested by users, the nearest PoP is contacted and if the content is available there, the request is served. The content can be present at the same PoP if content was first uploaded there or was brought there by some other request. If the content is not available, then a *pull* request is made and the content is brought to the PoP and served. This is the defacto mechanism (also known as a cold-miss) used by most CDNs [you]. We use this "serve-if-available" else "*pull*-when-not available" mechanism as the baseline and we shall show that this scheme can lead to high bandwidth costs. An example of this architecture is shown in Figure 3 where there are multiple interconnected PoPs around the world each serving a local user group.

4.2.2.2 Why is TailGate necessary: Toy Example

Consider user Bob living in Boston and assigned to the Boston PoP in Figure 3. Bob likes to generate and share content (videos, photos) with his friends and family. Most of Bob's social contacts are

geographically close to him, but he has a few friends on the West Coast US, Europe and Asia. These geographically distributed set of friends are assigned to the nearest PoP respectively. Bob logs in to the application at 6PM local time (peak time) and uploads a family video shot in HD that he wants to share. Like Bob, many users perform similar operations. A naive way to ensure this content to be as close as possible to all users before any accesses happen would be to *push* the updates/content to other PoPs immediately, at 6PM. Aggregated over all users, this process of pushing immediately can lead to a traffic spike in the upload link. Worse still, this content may not be consumed at all thus having contributed to the spike unnecessarily. Alternatively, instead of pushing data immediately, we can wait till the first friend of Bob in each PoP accesses the content. For instance Alice, a friend of Bob's in London logs in at 12PM local time and requests the content, and the system triggers a *pull* request, pulling it from Boston. However, user activity follow strong diurnal trends with peaks (12PM London local), hence multiple requests by different users will lead to multiple pulls, leading to yet another traffic spike. The problem with caching long-tailed content is well documented [ASKF10], and this problem is further exacerbated when Alice is the only friend of Bob's in London interested in that content and there are many such Alices. Hence all these "Alices" will experience a low QoE (as they have to wait for the content to be downloaded) and the provider experiences higher bandwidth costs – a loss for all.

TailGate's Approach: Instead of pushing content as soon as Bob uploads, wait till 2AM Boston local time, which is off-peak for the uplink, to push the content to London where it will be 7AM local time, again off-peak for downlink in London, and 7AM is still earlier that 12PM when Alice is likely to log in. Therefore Alice can access Bob's content quickly, hence experience relatively high QoE. The provider has transferred the content during off-peak hours, decreasing costs – a win-win scenario for all. TailGate is built upon this intuition where such time differences between content being *uploaded* and content being *accessed* is exploited. In a geo-diverse system, such time differences exist anyway. However, in order to exploit these time differences, TailGate needs information about the social graph (Alice is a friend of Bob), where these contacts reside (Alice lives in London) and the likely access patterns of Alice (she will likely access it at 12PM).

4.2.2.3 System Requirements

TailGate needs to address and balance the following requirements:

Reduce bandwidth costs: Despite the dropping price of leased WAN bandwidth and networking equipment, the growth rate of UGC combined with the incorporation of media rich long tail content (e.g. images and HD videos) makes WAN traffic costs a big concern [Ham, For]. For instance, the traffic volume produced by photos on Facebook can be in thousands of GB from just one region, e.g. NYC [WPD10]. This problem was handled in [LSYR11].

Decrease latency: The latency in the architecture described is due to two factors: one is the latency component in the access link between the user to the nearest PoP. The other component lies in getting that content from the source PoP, if the content is not available in the nearest PoP. Since the former is beyond our reach we focus on getting the content to the closest PoPs.

Online and reactive: The scale of UGC systems [hig] can lead to thousands of transactions per second as well as a large volume of content being uploaded per second. In order to handle such volume any solution has to be online, simple and react quickly.

4.2.2.4 What TailGate does not do

TailGate optimizes for bandwidth costs and does not consider storage constraints. It would be interesting to consider storage as well but we believe the relatively lower costs of storage puts the emphasis on reducing bandwidth costs [you]. TailGate does not deal with dynamic content like profile information etc. in OSNs as other systems do [PES10]. TailGate deals with large static UGC that is long-tailed and hence not amenable to existing distribution solutions.

4.2.3 Specifications

In this section, we first formulate the central problem that TailGate deals with – scheduling content updates to different PoPs in order to minimize bandwidth costs and latency, which we capture by way of a metric called "penalty". We then describe the algorithm TailGate uses that satisfies the requirements mentioned in section 4.2.2.3, along with existing solutions.

4.2.3.1 Formulation

Let $\{u_1,\ldots,u_N\}$ be the set of users and $\{S_1,\ldots,S_K\}$ the set of PoP sites, distributed at different locations around the world. As already discussed in section 4.2.2.1, we assume each user is assigned to a site and the user's content is uploaded to this site. The friends of this user can be assigned to this site or to the other sites, depending on their location. The social network is captured by the set $F(u_n)$, consisting of social contacts of user u_n. We denote by $S(u_n)$ the master(closest) site of user u_n, $u_n \rightarrow S_k$ the fact that the user u_n needs to send data to site S_k.

We model the evolution of the system at discrete time bins $t \in [0,T]$ and we assume that bins are short enough – typically a minute – so that a user performs at most one read and at most one update during a time bin. Updates and reads performed by user u_n during time bin t are denoted respectively by $w_n^{[t]}$ and $r_n^{[t]}$ (binary matrices). We denote by $\mathbf{w}_n^{[t]}$ the actual size of the updates sent by user u_n during time bin t. We assume that upon a read operation a user can access the content posted by the user's friends. Table 2 summarizes the terminology used in our formulation.

The decision variable of the optimization problem latency vs. bandwidth costs is the update schedule between sites, denoted by $t_{n,k}^{[t]}$; the number of updates of user u_n sent to site S_k during time bin t. We denote by $\mathbf{t}_{n,k}^{[t]}$ the size of the updates of user u_n sent to site S_k during time bin t. Transmission of updates of a *given* user to a given S_k follows a FIFO policy to ensure temporal consistency. Hence the missing updates are always the most recent ones.

Optimization metric: Bandwidth costs The incoming and outgoing traffic volumes of each site S_k depends on the upload strategy and updates:

$$v_k^{\downarrow[t]} = \sum_{k' \neq k} \left(\sum_{u_n \rightarrow S_{k'}} \mathbf{t}_{n,k}^{[t]} \right) \quad (1)$$

$$v_k^{\uparrow[t]} = \sum_{k' \neq k} \left(\sum_{u_n \rightarrow S_k} \mathbf{t}_{n,k'}^{[t]} \right) \quad (2)$$

In general, a peak-based pricing scheme is used as a cost function ($p_k(\cdot)$). The most common is the 95th percentile ($q(\cdot)$) of the traffic volume (typically a linear function whose slope depends on the location of the site, *i.e.*, bandwidth prices vary from one city to another). Therefore the bandwidth costs incurred at site S_k is $c_k = p_k(\max(q(v_k^{\downarrow}), q(v_k^{\uparrow})))$ and the total bandwidth cost is the sum of all the c_k.

Constraint: Latency via Penalty metric In order to capture the notion of latency, which is closely related to a "cold-miss" at a site (as discussed in section 4.2.2.1), for a user u_n at site S_k is captured by the number $d_{n,k'}^{[t]}$; updates of u_n that are missing at site S_k:

$$d_{n,k}^{[t]} = \begin{cases} \sum_{t'=0}^{t'=t} w_n^{[t]} - t_{n,k}^{[t]} & \text{if} S(u_n) \neq S_k \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

which is representative of the number of times the content has to be fetched from the server where it is originally hosted, increasing latency. To evaluate the perceived latency, we define a penalty system: every time a user requests one of her friends' updates and it is not available, the total penalty is incremented by the number above. The total penalty P is:

$$P(T) = \sum_{u_n \in \cup} \left(\sum_{u_m \in F(u_n)} d_{n,S(u_m)}^{[t]} \cdot r_m^{[t]} \right) \quad (4)$$

Handling real-world constraints: In the online case, reads are not known in advance and the algorithm must therefore be oblivious of the input read matrix $r_n^{[t]}$. Moreover, an OSN might be hesitant to release individual read patterns and the social graph to the entity (CDN) operating TailGate. For these reasons, we replace the reads in the above problem with estimated reads – a generic diurnal pattern. Assuming that each user of the social network has a regular read activity, captured by a probability $\rho_n^{[t]}$ of user u_n performing a read operation during time bin t, then the probability of an update posted at time t by user u_n to be read at time t' on server S_k: $\rho_{n,k}^{[t]} = 1 - \prod_{u_m \to F_n \cap S_k} (1 - \rho_m^{[t]})$. We can therefore derive an expression of the expected read, which is the deadline $\delta_{n,k}^{[t]}$ (*i.e.*, first read) on server S_k of an update posted by u_n at time t and this replaces $r_n^{[t]}$:

$$\delta_{n,k}^{[t]} = \sum_{t'=t}^{+\infty} \left(t' \cdot \rho_{n,k}^{[t']} \cdot \prod_{t''=t}^{t'-1} (1 - \rho_{n,k}^{[t']}) \right). \quad (5)$$

T	Number of time intervals in a charging period, $t \in T$.
$F(u_n)$	Set of nodes in social network of user u_n.
$w_n^{[t]}$	Update pattern of user u_n during interval t (binary). $\mathbf{w}_n^{[t]}$ the update traffic volume.
$r_n^{[t]}$	Read pattern of user u_n during interval t (binary).
$t_{n,k}^{[t]}$	Schedule of u_n's updates to site S_k during interval t. $\mathbf{t}_{n,k}^{[t]}$ the update traffic volume.
$v_k^{\downarrow[t]}$	Incoming traffic volume at site S_k
$v_k^{\uparrow[t]}$	Outgoing traffic volume at site S_k
p_k	Pricing function at site S_k

Table 2: TailGate Formulation Notation

Assuming, for the sake of simplicity, a pricing scheme based on the maximum utilization of the link (instead of the 95th percentile), it can be shown that the decision versions of optimization problem is NP-Complete. We can do a simple reduction from the *partition* problem to show this.

4.2.3.2 Heuristic

To keep TailGate simple, we resort to a greedy heuristic to schedule content. At a high level, we consider load on different links to be divided into discrete time bins (for instance, 5 min bins). Then, the heuristic is simple – given an upload (triggered by a write) at a given time at a given site that needs to be distributed to different sites, find or *estimate* the bin in the future in which this content will likely be read, and then schedule this content in the least loaded bin amongst the set of bins: (current bin, bin in which read occurs). If more than one candidate bin is found, pick a bin at random to schedule. Simultaneous uploads are handled randomly; no special preference is given to one upload over another.

We highlight the salient points of this approach: (i) This is an online scheme in the sense that content is scheduled as it is uploaded. (ii) This scheme optimizes for upload bandwidth only; we tried a greedy variant where we optimized for upload and download bandwidth, but did not see much improvement, so we settled for a simpler scheme. (iii) If we have perfect reads, TailGate produces no penalties by design. However, this won't be case and we quantify the tradeoff in the next section. (iv) In the presence of background traffic, one can use available bandwidth estimation tools to measure and forecast.

4.2.3.3 Existing Solutions

We describe two solutions – push/FIFO and a pull based approach that mimic various cache-based solutions (including CDNs) that can be used to distribute long-tailed content.

For all the schemes we consider, we assume storage is cheap and once content (for instance a video) is delivered to a site, all future requests for that content originating from users of that site will be served locally. In other words, content is moved between sites only once. Flash-crowd effects etc. are therefore handled by the nearest PoP. The key difference between the schemes we consider, is *when* the content is delivered.

Immediate Push/FIFO: The content is distributed to different PoPs as soon as it is uploaded. Assuming there are no losses in the network, FIFO decreases latency for accesses as content will always be served from the nearest PoP.

Pull: The content is distributed only when the first read request is made for that content. This scheme therefore depends on read patterns and we use the synthetic reads to figure out the first read for each upload. Note that in this scenario, the user who issues the first read will experience higher latency.

4.2.3.4 Deployment Scenarios

OSN running TailGate: An OSN provider like Facebook can run TailGate. In this case, all the necessary information can be provided and TailGate provides the maximum benefit. The distributed architecture we have considered throughout is different from that employed currently by Facebook that operates three datacenters, two on the west coast (CA) and one on the eastern side (VA) and leases space at other centers [Kno]. The VA datacenter operates as a slave to the CA datacenters and handles traffic from east coast US as well as Europe. All writes are handled by a datacenter in CA. However, we believe that large OSNs will eventually gravitate to the distributed architecture we described in section 4.2.2.1, for the reasons of performance and reliability mentioned in section 4.2.1 as well as recent work that has shown that handling reads/writes out of one geographical site can be detrimental to performance for an OSN [WPD10], pointing to an architecture that relies on distributed state. If the OSN provider leases bandwidth from external providers, Tailgate decreases costs. If the provider owns the links, then Tailgate makes optimal use of the link capacity – delaying equipment upgrades as networks are normally provisioned for the peak.

CDNs with social information: Systems like CDNs are in general highly distributed (for instance Akamai), but the architecture we used in this work captures fundamental characteristics like users being served out of the nearest PoP [HWLR08]. Existing CDN providers may not get access to social information, yet may be used by existing OSN providers to handle content (this is changing [Ken]). We have shown that even with limited access, the CDN provider can still optimize for bandwidth costs after making assumptions about the access patterns.

CDNs without social information: Even without access to OSN information, a CDN can access publicly available information (like Tweets) and use that to improve performance for its own customers.

4.2.3.5 Related Work

Distribution of long-tailed content has been addressed by several works, but most of the work has been confined to distribution of such content on P2P networks [PS09, MRL09]. However, such swarm systems need extra resources (by way of replicates), and as such do not address transit bandwidth costs or latency constraints explicitly – requirements that Tailgate addresses.

The popularity of OSNs has led to work that exploits social networking information to better inform system design. TailGate is, in part, motivated by findings presented by Wittie et al [WPD10] where the authors analyze the current Facebook architecture and uncover network performance problems the architecture introduces, including high bandwidth utilization and large delays. The notion of distributing state to improve performance based on geography or via clustering users on a social graph has been explored by others as well [PER09, KGNR10]. Tailgate can be seen complimentary to these solutions as the underlying goals – reduce bandwidth costs, and reduce end-user latency are similar.

A related work is Buzztraq that proposes to use "social-cascades" to trigger content distribution [SYC09]. TailGate is similar to Buzztraq, in that it relies on social information, however, TailGate does not need "cascades" to occur to inform content distribution. In that sense TailGate is much simpler and yet effective. There has been recent work that combines information from OSNs to improve CDN service [SMMC11], and hence is similar in motivation with TailGate. The authors propose a similar mechanism to Buzztraq wherein social cascades can be used to place content close to users. TailGate also aims to place content close to users, however our focus is *when* to distribute such content to minimize bandwidth costs. TailGate can be used along with the approach proposed in [SMMC11] that answers *where*. Work by Laoutaris et al [LSYR11] describes a system called NetStitcher that aims to do bulk transfers between datacenters, by exploiting off-peak hours and storage in the network to send bulk data. NetSticher operates at the network layer, and TailGate relies on information about diurnal access patterns at the application layer. Hence TailGate and NetSticher are complimentary.

5. CONCLUSION

This documents summarises the final refinements of the network-aware overlay application techniques built by the ENVISION project, including the distributed data management infrastructure and the interactive video distribution tree optimisation algorithm.

The cross-layer optimisation may involve in some scenarios trading off optimality at the overlay layer for a reduction in the costs incurred by the ISPs. A theoretical framework is developed for the study of the *cooperation utility*, a function that expresses this tradeoff in terms of the traffic volume, overlay quality and ISP cost associated with any particular overlay flow. The cooperation utility can be used to analyse the feasible operational boundaries for overlays and ISPs with minimum quality requirements and maximum cost restrictions respectively.

An overlay connection that is desirable by an ISP at the originating end of the traffic may be incurring additional costs for the ISP at the terminating end, or a connection that is ranked as a better alternative may be detrimental for the other ISP it involves. It quickly becomes evident that the simplistic approach of taking into consideration the preference of a single ISP for any overlay connection leads to suboptimal outcomes. An approach for addressing this issue is proposed in this document and it involves the use of voting schemes to allow for the consolidation of diverging and possibly conflicting sets of preferences provided by all the ISPs hosting overlay nodes.

Building on an hierarchical clustering structure of all Internet endpoints, an n-casting protocol is developed to enable the scalable and efficient indexing and querying of overlay resource information, with statistically bound errors regarding the accuracy of the query resolution processes. Resources are filtered using an identifier and ranked based on the network delay between the querying overlay node and the candidate resource. The *n* distinct best matches are returned.

A tree based content distribution system is developed for interactive video applications where bounded delay is a requirement and the number of participant nodes is relatively small, allowing for simple centralised implementations of the overlay coordination functions. The limited capacity provided by the user end systems (participant nodes) may result in the construction of a delivery tree violating the latency upper bound requirement associated with interactive video. A Tree Optimisation algorithm is designed for mitigating this problem by including High Capacity Nodes in strategic locations in a cost-effective manner, minimising the total cost associated with the usage of the High Capacity Node services provided by the network operators through CINA.

The increasing popularity of user-generated content and the rise of online social networks as a distribution mechanism has increased the demand for long-tailed content, i.e. content that is popular among small groups of users. TailGate is a content distribution optimisation technique that plans the content transmission to a particular destination based on network information about the cost of using that link over time and application information for predicting the content demand at particular locations and times.

This deliverable concludes the design and specifications work in WP4. The techniques specified here and in previous deliverables will be evaluated in WP6, and the final evaluation results will be published in D6.2 at the end of the year.

6. REFERENCES

[AAF08] Vinay Aggarwal, Obi Akonjang, and Anja Feldmann. Improving User and ISP Experience through ISP-aided P2P Locality. In Proc. of the Global Internet Symposium, 2008.

[ADJ+10] Sharad Agarwal, John Dunagan, Navendu Jain, Stefan Saroiu, and Alec Wolman. Volley: Automated Data Placement for Geo-Distributed Cloud Services. In NSDI, 2010.

[AFS07] Vinay Aggarwal, Anja Feldmann, and Christian Scheideler. Can ISPs and P2P users cooperate for improved performance? SIGCOMM Comput. Commun. Rev., 37:29-40, July 2007.

[ASKF10] Bernhard Ager, Fabian Schneider, Juhoon Kim, and Anja Feldmann. Revisiting Cacheability in Times of User Generated Content. In Global Internet, 2010.

[BCC+06] Ruchir Bindal, Pei Cao, William Chan, Jan Medved, George Suwala, Tony Bates, and Amy Zhang. Improving traffic locality in BitTorrent via biased neighbor selection. In Proc. of ICDCS '06, page 66, Washington, DC, USA, 2006.

[BLD10] Stevens Le Blond, Arnaud Legout, and Walid Dabbousa. Pushing bittorrent locality to the limit. Comput. Netw., 55(3), 2010.

[BV09] Stephen Boyd and Lieven Vandenberghe. Convex Optimization. Cambridge University Press, 2009.

[CB08] David R. Choffnes and Fabián E. Bustamante. Taming the torrent: a practical approach to reducing cross-ISP traffic in peer-to-peer systems. In Proc. of SIGCOMM '08, pages 363-374, USA, 2008. ACM.

[CD28] Charles W. Cobb and Paul H. Douglas. A theory of production. The American Economic Review, 18(1):139-165, March 1928.

[CYRK03] Casey Carter, Seung Yi, Prashant Ratanchandani, and Robin Kravets. Manycast: exploring the space between anycast and multicast in ad hoc networks. In MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking, pages 273–285, New York, NY, USA, 2003. ACM.

[D3.2] ENVISION deliverable D3.2, Refined Specification of the ENVISION Interface, Network Monitoring and Network Optimisation Functions Initial, December 2011, FP7 ICT ENVISION project, www.envision-project.org

[D3.3] ENVISION deliverable D3.3, Final Specification of the ENVISION Interface, Network Monitoring and Network Optimisation Functions, June 2012, FP7 ICT ENVISION project, www.envision-project.org

[D4.1] ENVISION deliverable D4.1, Initial Specification of Consolidated Overlay View, Data Management Infrastructure, Resource Optimisation and Content Distribution Functions, December 2010, FP7 ICT ENVISION project, www.envision-project.org

[D4.2] ENVISION deliverable D4.2, Refined Specification of Consolidated Overlay View, Data Management Infrastructure, Resource Optimisation and Content Distribution Functions, December 2011, FP7 ICT ENVISION project, www.envision-project.org

[D6.1] ENVISION deliverable D6.1, Initial Testbed Description and Preliminary Evaluation Results of Content-aware Cross-layer Optimizations for Advanced Multimedia Applications, December 2011, FP7 ICT ENVISION project, www.envision-project.org

[DHKS09] X. Dimitropoulos, P. Hurley, A. Kind, and M. P. Stoecklin, "On the 95-percentile billing method," in Proc. of PAM. Springer-Verlag, pp. 207–216.

[DLL+11] Jie Dai, Bo Li, Fangming Liu, Baochun Li, and Hai Jin. On the efficiency of collaborative caching in ISP-aware p2p networks. In Proc. of INFOCOM, pages 1224-1232. IEEE, 2011.

[Dun92] R. I. M. Dunbar. Neocortex Size as a Constraint on Group Size in Primates. Journal of Human Evolution, 22(6):469–493, 1992.

[F07] P. Faratin, "Economics of overlay networks: An industrial organization perspective on network economics," in Proceedings of NetEcon, 2007.

[Faca] Facebook. Facebook Ranked Second Largest Video Site. http://vator.tv/news/2010-09-30-facebook-ranked-second-largest-video-site.

[Facb] Facebook. Facebook User Statistics. http://www.facebook.com/press/info.php?statistics.

[For] Forrester Consulting. The Future of Data Center Wide Area Networking. http://www.infineta.com/news/news_releases/press_release: 5585,15851,446.

[GR04] Hugh Gravelle and Ray Rees. Microeconomics. Prentice Hall, 3rd edition, 2004.

[Ham] James Hamilton. Inter-Datacenter Replication and Geo-Redundancy. http://perspectives.mvdirona.com/2010/05/10/InterDatacenterReplicationGeoRedundancy.aspx.

[HWLR08] Cheng Huang, Angela Wang, Jin Li, and Keith W. Ross. Measuring and Evaluating Large-Scale CDNs. In IMC, 2008.

[JZSRC08] Wenjie Jiang, Rui Zhang-Shen, Jennifer Rexford, and Mung Chiang. Cooperative content distribution and traffic engineering. In Proc. of NetEcon, 2008.

[KGNR10] Thomas Karagiannis, Christos Gkantsidis, Dushyanth Narayanan, and Antony Rowstron. Hermes: Clustering Users in Large-Scale E-mail services. In SoCC, 2010.

[KMK+09] N. Kamiyama, T. Mori, R. Kawahara, S. Harada, and H. Hasegawa. ISP-Operated CDN. In Proc. of the Global Internet Symposium, 2009.

[Kno] DataCenter Knowledge. Facebook data center faq. http://www.datacenterknowledge.com/the-facebook-data-center-faq/.

[Lin] Greg Linden. Marissa Mayer at Web 2.0. http://glinden.blogspot.com/2006/11/marissa-mayer-at-web-20.html.

[LMC+12] Raul Landa, Eleni Mykoniati, Richard G. Clegg, David Griffin, and Miguel Rio, Modelling the Tradeoffs in Overlay-ISP Cooperation, to appear in the proceedings of IFIP Networking 2012.

[LMG+12] R. Landa, E. Mykoniati, D. Griffin, M. Rio, N. Schwan, I. Rimac, Overlay Consolidation of ISP-Provided Preferences, Proceedings of the International Workshop on Cross-Stratum Optimization for Cloud Computing and Distributed Networked Applications, July 2012, Madrid.

[LSRS09] N. Laoutaris, G. Smaragdakis, P. Rodriguez, and R. Sundaram, "Delay tolerant bulk data transfers on the Internet," in Proc. of ACM SIGMET-RICS, 2009.

[LSYR11] Nikolaos Laoutaris, Michael Sirivianos, Xiaoyuan Yang, and Pablo Rodriguez. Inter-Datacenter Bulk Transfers with NetStitcher. In SIGCOMM, 2011.

[MCL+07] R. T. B. Ma, D. M. Chiu, J. C. S. Lui, V. Misra, and D. Rubenstein, "Internet economics: the use of Shapley value for ISP settlement," in Proceedings of CoNEXT, 2007, pp. 1–12.

[MDGV11] M. Marcon, M. Dischinger, K. Gummadi, and A. Vahdat, "The local and global effects of traffic shaping in the internet," in Proc. of IEEE COMSNETS, Jan. 2011, pp. 1–10.

[MRL09] Daniel S. Menasche, Antonio A.A. Rocha, Bin Li, Don Towsley, and Arun Venkataramani. Content Availability and Bundling in Swarming Systems. In CoNEXT, 2009.

[OSL12] Otto J., Stanojevic R., Laoutaris N., Temporal Rate Limiting: cloud elasticity at a flat fee, NetEcon 2012

[PER09] Josep M. Pujol, Vijay Erramilli, and Pablo Rodriguez. Divide and Conquer: Partitioning Online Social Networks. http://arxiv.org/abs/0905.4918, 2009.

[PES10] Josep M. Pujol, Vijay Erramilli, Georgos Siganos, Xiaoyuan Yang, Nikolas Laoutaris, Parminder Chhabra, and Pablor Rodriguez. The Little Engines that Could: Scaling Online Social Networks. In SIGCOMM, 2010.

[PFA+10] Ingmar Poese, Benjamin Frank, Bernhard Ager, Georgios Smaragdakis, and Anja Feldmann. Improving content delivery using provider-aided distance information. In Proc. of IMC '10, pages 22-34, USA, 2010. ACM.

[PMG09] Jon Peterson, Enrico Marocco, and Vijay Gurbani. Application-Layer Traffic Optimization (ALTO) working group, 2009.

[PS09] Ryan S. Peterson and Emin Gün Sirer. AntFarm: Efficient Content Distribution with Managed Swarms. In NSDI, 2009.

[RLY+11] Rubén Cuevas Rumín, Nikolaos Laoutaris, Xiaoyuan Yang, Georgos Siganos, and Pablo Rodriguez. Deep diving into bittorrent locality. In Proc. of INFOCOM, pages 963-971. IEEE, 2011.

[SCG11] R. Stanojevic, I. Castro, and S. Gorinsky, “CIPT: using tuangou to reduce IP transit costs,” in Proc. of ACM CONEXT. ACM, 2011

[SCPR09] Marco Slot, Paolo Costa, Guillaume Pierre, and Vivek Rai. Zero-day reconciliation of bittorrent users with their ISPs. In Proc. of Euro-Par 15, pages 561-573, Berlin, Heidelberg, 2009. Springer-Verlag.

[SFKW09] Fabian Schneider, Anja Feldmann, Balachander Krishnamurthy, and Walter Willinger. Understanding Online Social Network Usage from a Network Perspective. In IMC, 2009.

[SMMC11] Salvatore Scellato, Cecilia Mascolo, Mirco Musolesi, and Jon Crowcroft. Track Globally, Deliver Locally: Improving Content Delivery Networks by Tracking Geographic Social Cascades. In WWW, 2011.

[SYC09] Nishanth Sastry, Eiko Yoneki, and Jon Crowcroft. Buzztraq: Predicting Geographical Access Patterns of Social Cascades Using Social Networks. In SNS, 2009.

[TFK11] Ruben Torres, Alessandro Finamore, Jesse Kim, Marco Mellia, Maurizio M. Munafo, and Sanjay Rao. Dissecting Video Server Selection Strategies in the YouTube CDN. Technical Report TR-ECE-11-02, Purdue University, 2011.

[Twi] Twitter. Growing Around the World. http://blog.twitter.com/2010/04/growing-around-world.html.

[urlb] Facebook Hosts More Photos than Flickr and Photobucket. http://www.tothepc.com/archives/facebook-hosts-more-photos-than-flickr-photobucket/.

[WPD10] Mike P. Wittie, Veljko Pejovic, Lara Deek, Kevin C. Almeroth, and Ben Y. Zhao. Exploiting Locality of Interest in Online Social Networks. In CoNEXT, 2010.

[XYK+08] Haiyong Xie, Y. Richard Yang, Arvind Krishnamurthy, Yanbin Grace Liu, and Abraham Silberschatz. P4P: Provider portal for applications. SIGCOMM Comput. Commun. Rev., 38(4):351-362, 2008.

[you] YouTube CDN Architecture. Private Communication, Content Delivery Platform, Google.