Enriched Network-aware Video Services over Internet Overlay Networks

www.envision-project.org

Deliverable D4.2

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

Public report, Version 1, 27 February 2012

Authors

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

ALUD Nico Schwan, Ivica Rimac, Klaus Satzke

LaBRI Toufik Ahmed, Abbas Bradai, Ubaid Abassi, Samir Medjiah

TID Nikolaos Laoutaris, Oriol Ribera Prats

LIVEU Noam Amram

Reviewers Bertrand Mathieu, Noam Amram

Abstract This document elaborates on the modelling of the tradeoff between application optimality and ISP cost and specifies techniques for consolidating preferences across different ISPs. A scalable resource information management system is developed providing n-casting capabilities at the application layer. Application-specific content distribution techniques are proposed for three types of content: live, interactive and long-tailed static content. Each of these techniques explores different aspects of the collaboration between the application and the underlying network, including the use of information for the ISP costs and the integration of network capabilities and ISP resources offered through CINA into the overlay application in a dynamic and cost-effective manner. Finally, a comprehensive approach is proposed, exploring a practical application of overlay monitoring information together with the CINA costs in order to improve the operation of a CDN overlay network.

Keywords Network-aware Content Distribution, Cross-layer Optimisation, Consolidated Overlay View, Distributed Resource Data Management, Live and Interactive Video, Online Social Network, CDN

© Copyright 2012 ENVISION Consortium

University College London, UK (UCL)

Alcatel-Lucent Deutschland AG, Germany (ALUD)

Université Bordeaux 1, France (LaBRI)

France Telecom Orange Labs, France (FT)

Telefónica Investigacion y Desarollo, Spain (TID)

LiveU Ltd., Israel (LIVEU)

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 second 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 [D3.2].

The information provided by the ISPs through the CINA interface may reflect objective performance metrics that are independent or the preferences of the ISPs determined based on their particular business policies and optimisation objectives. In the first case, several possibilities are identified which may lead to increasing accuracy of the overlay network performance models. 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.

Application-specific content distribution techniques are proposed for three types of content: live, interactive and 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. The work on the live content distribution is aligned with the specifications of the content adaptation components documented in [D5.2], and will lead to a joint prototype between the topology construction algorithms developed in WP4 and the layer smoothing and chunk request scheduling algorithms developed in WP5.

Beyond addressing the technical challenges related with their corresponding applications, each of these techniques explores different aspects of the collaboration between the application and the underlying network, including the use of information for the ISP costs and the integration of network capabilities and ISP resources offered through CINA into the overlay application in a dynamic and cost-effective manner. Finally, a comprehensive approach is proposed, exploring a practical application of overlay monitoring information together with the CINA costs in order to improve the operation of a CDN overlay network.

The feedback received from the parallel evaluation activities in WP6 documented in [D6.1] will be used to refine the specifications in this document. The work in WP4 will conclude at M30 (June 2012) with a software release, the finalisation and documentation of the protocol and algorithm specifications in D4.3.

Copyright © ENVISION Consortium, February 2012

TABLE OF CONTENTS

Copyright © ENVISION Consortium, February 2012

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 first phase of the work, captured in [D4.1], focussed on analysing the requirements of particular applications, identifying the research challenges and defining the functions at the overlay layer and the interactions between them. In this document the focus is on providing analytical models and specifications for the algorithms, protocols and subsystems as applicable to the particular function under consideration and its research challenges and evaluation possibilities.

The consolidation of the information received through CINA and of the application layer information is the consideration of the Consolidated Overlay View functionality. While in many scenarios overlay application and ISP network optimisation objectives are perfectly aligned, there are cases where the overlay application will need to include additional considerations when determining its overlay topology and traffic matrix. To study the relationship between the overlay optimality and the ISP costs, in section 2.2 we propose a model for the related tradeoff optimisation functions that are applicable in the general case.

Based on their interconnection types, traffic demand patterns and transit pricing models, different ISPs will have different interests regarding sending and receiving traffic from particular destinations at particular times of the day, communicating to the overlay diverging or possibly also conflicting sets of preferences. Techniques based on voting systems are explored in section 2.3 for consolidating these subjective views on the desirability of a particular overlay connection from the perspective of the network operators involved to a single value that can be used by the overlay.

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 a large number of content items, in a way that it scales with the size of the overlay and is 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.

Application-specific content distribution techniques are proposed for three types of content: live, interactive and 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, see sections 4.1, 4.2 and 4.3 respectively. Beyond addressing the technical challenges related with their corresponding applications, each of these techniques explores different aspects of the collaboration between the application and the underlying network, including the use of information for the ISP costs and the integration of network capabilities and ISP resources offered through CINA into the overlay application in a dynamic and cost-effective manner. Finally, a comprehensive approach is proposed in section 4.4, exploring a practical application of overlay monitoring information together with the CINA costs in order to improve the operation of a CDN overlay network.

While some of these specifications are finalised, others are in an intermediate stage and will be further refined in the following reporting period. The evaluation specifications and preliminary results corresponding to the specifications described in this document can be found in [D6.1].

Copyright © ENVISION Consortium, February 2012

2. CONSOLIDATED OVERLAY VIEW FUNCTIONS

The following sections elaborate on three different aspects of consolidating information received through CINA. First, we are discussing the possibilities for ISPs and overlays to exchange through CINA and consolidate information which represents network performance metrics. This work aims at identifying uses for the interface and remains at a theoretical exploratory level. Second, 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.

2.1 Collaborative Network Performance Modelling

This section elaborates on the possible forms of collaboration between overlay applications and ISPs with the purpose of improving the overlay network performance modelling functions, either by increasing their accuracy or by reducing their load.

2.1.1 Intra-domain Information for Metrics that can be Spatially Composed

Spatial composition of metrics is defined in [RFC2330] and encompasses the definition of end-to-end performance metrics based on metrics collected on segments of the end-to-end path, including for example delay, loss, available bandwidth etc. and excluding throughput. For more information on spatial composition, see also [IPPM-SC16].

ISPs could provide information about the performance in the path segment from the ingress router to the egress router in their domain and the overlay could use the available formulas for calculating the information obtained from all the ISPs in a path to extract the end-to-end value. This approach, however, has several disadvantages. Firstly, it requires that the ISPs reveal information about the performance in their domains at a level of detail that is typically considered of confidential nature. Secondly, it requires transit ISPs to implement the CINA interface and interact with applications that have no presence in their domain.

The overlay application could perform additional end-to-end measurements to compensate for the lack of cooperation from transit ISPs or in general ISPs that are not ENVISION-enabled. Several techniques already exist for creating a path model and using end-to-end measurements with measurements on path segments. A prime example of a system implementing this technique is iPlane [MIP+06, MAK06], which uses measurements to create an “atlas” of the Internet clustered on the basis of BGP atoms [Bkc01] (minimal elements experiencing equivalent routing paths). iPlane estimates an abstracted view of the route that traffic will take between two end points in terms of route sections. It then assesses delay loss rate, capacity and available bandwidth (unused capacity) on route sections and hence for the entire route. This system could be significantly enhanced by using information from the ISPs to fine-tune the inference of the route sections and the network performance for these path segments that are controlled by ENVISION-enabled ISPs.

2.1.2 ISP End-to-End Performance Modelling

A possible cooperation scenario between the application and the ISP could involve offloading the network performance estimation to the ISP and reducing the role of the application to only collect overlay measurements, see Figure 1.

ISPs are in an advantageous position to operate observation points and collect traces for traffic originating or terminating at their domain. In addition to explicitly measuring network performance at the network layer, information about the network conditions can be inferred by looking at transport layer information and in particular TCP. TCP sessions, especially retransmissions and the SYN-ACK packets, can be used to estimate loss and RTT [KAPA91, BOLO93]. Further, ISPs providing

Copyright © ENVISION Consortium, February 2012

content servers or other application-layer services accessible from local and remote IP addresses, can collect more passive measurements for the network performance beyond the boundaries of their domain. Finally, ISPs often collect flow-level information using tools like NetFlow, which can be useful to derive statistics regarding the volume and the type of traffic originating or terminating at certain remote destinations.

This information together with overlay passive measurements provided by all ENVISION-enabled applications can be used to derive models about the network performance beyond the ISP domain, building a wider and statistically more accurate view, taking into account traffic from a large number of remote locations and time of the day variations. These models, as they are derived from end-to-end measurements are not just limited to metrics that can be spatially composed, but include also metrics like throughput. Applications could then enquiry for the end-to-end delay between PIDs, as calculated by the prediction models derived by the ISP.

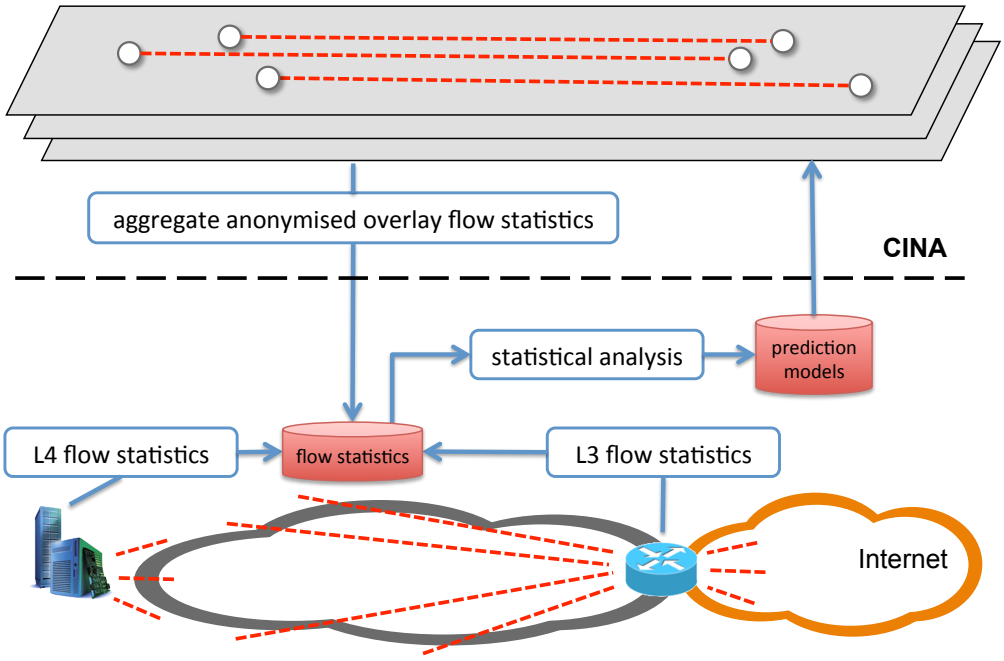

Figure 1: Network Performance Modelling as a CINA Service

This scenario raises some confidentiality issues regarding the exposure of overlay measurements to the ISP and some trust issues regarding the network performance predictions generated by the ISP. These issues could be addressed with appropriate anonymisation of the overlay measurements and with auditing functions performing regular and targeted verification of the network performance predicted by the ISP.

Another case in which CINA could successfully contribute to a better performance estimation is by providing accurate geolocation information, that can be successfully used to increase the accuracy of network coordinate systems. For instance, in [AGLO09] Agarwal and Lorch propose a system that predicts latencies between machine pairs, allowing a network gaming platform to cluster users into sessions that have low latency to each other. This system, *Htrae*, synthesises geolocation with a network coordinate system. It uses geolocation to select reasonable initial network coordinates for new machines joining the system, allowing it to converge more quickly than standard network coordinate systems and produce substantially lower prediction error than state-of-the-art latency prediction systems.

Copyright © ENVISION Consortium, February 2012

2.1.3 Endpoint Aggregation based on Network Performance

The collaboration between application and ISP could be beneficial in terms of reducing the monitoring load at the overlay layer. The ISPs aggregate the local (and in some cases possibly also the remote) endpoints to groups based on topological, routing, load, or other historical network performance information. The overlays perform measurements between endpoints only when there are no existing current measurements for the groups they belong to.

Aggregating endpoints using topology and routing information, allows the ISP to withhold some information and increases the efficiency of the transactions over the CINA interface. The burden for discovering the network performance lies entirely with the overlay, and the contribution of the ISP is simply to increase the efficiency of this process.

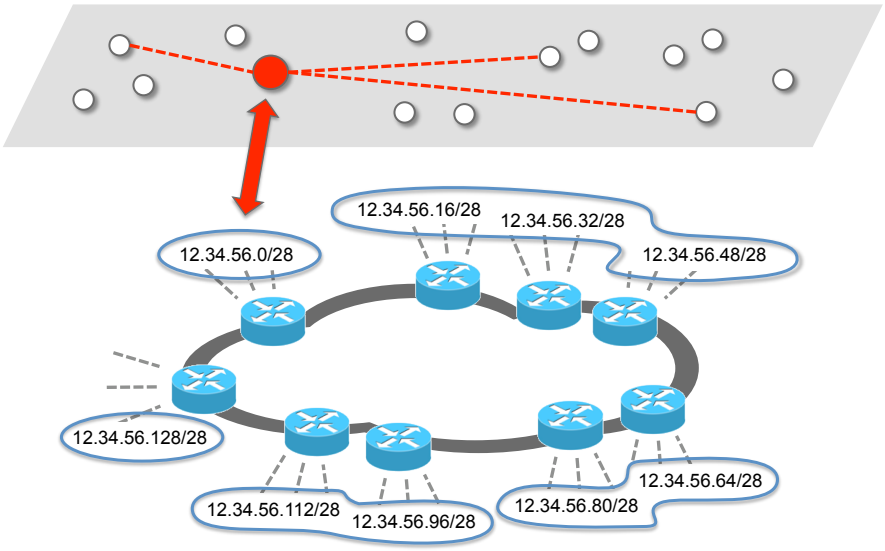

Figure 2: Node Clustering in CINA based on Network Performance

2.2 Overlay-ISP Cooperation Tradeoff

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.

The rest of this section has been suppressed from the public version of this deliverable. A full description of the Overlay-ISP Cooperation Tradeoff work is available at [LMC+12].

2.3 ISP Preference Consolidation

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

Copyright © ENVISION Consortium, February 2012

[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. 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.

The rest of this section has been suppressed from the public version of this deliverable. A full description of the ISP Preference Consolidation work is available at [LMG+12].

3. DISTRIBUTED DATA MANAGEMENT INFRASTRUCTURE FUNCTIONS

In ENVISION, a unique identifier is used for each type of overlay resource, e.g. content relaying nodes, content adaptation nodes, idle 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 to define 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 a node that is registered membership with a group identifier and that 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].

In ENVISION, we develop 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.

Copyright © ENVISION Consortium, February 2012

4. OVERLAY RESOURCE OPTIMISATION AND CONTENT DISTRIBUTION

This section presents application-layer techniques and functions for resource optimisation and content distribution. In particular, section 4.1 elaborates on techniques for building overlay topologies that result in good performance for live video distribution, while also incorporating resources provided by the ISP. Section 4.1.3.4 describes a system for managing a tree overlay topology for the distribution of interactive video. Section 4.3 presents a system that uses information from online social networks to predict the demand for content and proactively schedule the content replication at times when the cost for the ISP is the lowest. A comprehensive approach for selecting a CDN node based on proximity using network information obtained through CINA and overlay measurements is presented in section 4.4. Finally, section 4.5 elaborates on a technique that invokes ISP resources following the fluctuations of the application demand and number of users, while also ensuring a maximum cost for the invoked resources over a given time period.

4.1 Live Video Content Distribution

4.1.1 Problem Statement

This section focuses on the design of an overlay topology construction algorithm for live content with high bandwidth requirements and layered encoding. This algorithm will be used to investigate the performance gains achieved with the mobilisation of ISP and participant resources, and in particular high-capacity nodes and multicast transmission at individual ISPs.

The overlay topology construction algorithm needs to satisfy the following requirements:

- Support content with different bitrates and various layered encoding profiles.
- Support large numbers of nodes distributed at arbitrary locations in the world.
- Support nodes with various capabilities and different requirements for the quality level they receive.
- Support various levels of upload capacity made available to the overlay at each consumer node.
- Optimise the stream quality for each consumer node, i.e. the quality of the video that can be sustainably received at any point in time, in terms of spatial, temporal and video signal resolution.
- Optimise the stream liveness for each consumer node, i.e. the delay between a video data unit being sent by the video source node and its rendering by the consumer.
- Integrate high-capacity nodes in the overlay topology and determine their stream layer replication policies to maximise the benefit for the overlay.
- Integrate multicast service capabilities in the overlay topology for groups of nodes that belong to the same ISP.

4.1.2 Approach

We are approaching the problem of overlay topology construction for live video streaming and layered SVC encoding with the use of heuristics. Starting from a simple method, where nodes are added to the overlay one at a time and taking into account only their own properties, we are proceeding with a more sophisticated method where the nodes surrounding a node also play a role on determining its position in the overlay, and we are completing our study with a method that is closer to a fully distributed implementation with distinct policies for the selection of receiver and the selection of a sender.

Copyright © ENVISION Consortium, February 2012

These techniques are then enhanced to take into account the availability of participant and ISP upload capacity resources. Specialised replication policies are developed, taking into account the demand for a particular content quality layer at a particular area. Further, the support for multicast transmission inside the domain of some ISPs is also integrated in the overlay topology construction algorithms, enabling the evaluation of the corresponding performance gains.

As a first step, we are assuming complete information and freedom in establishing connections between any pair of nodes in the overlay. Using these results as a benchmark, we proceed with considering policies for creating clusters of nodes and modifying the overlay topology construction policies to operate within and across the clusters.

The results of this theoretical analysis will be fed to a prototype developed jointly with the partners active in WP5.

4.1.3 Specifications

4.1.3.1 Overlay Topology Construction

The overlay topology construction algorithms considered here aim at building an efficient topology for a given set of nodes with a given set of capabilities and preferences in terms how many quality layers they want to receive. Each node is assigned with a given upload capacity and its delay from any other node in the overlay is known. The overlay topology construction algorithm involves two basic steps: assignment of upload capacity to layers per node and creation of overlay connections per layer. The following sections describe these two policies in detail. Although at this point the capacity allocation policy is decoupled from the overlay connection selection policy, an approach that combines these two aspects under a unified policy is left for future work.

4.1.3.1.1 Upload Capacity Allocation to Layers

For a layered encoding technique like SVC that we are considering here, higher layers depend on lower layers and as a result a larger number of nodes will be subscribing to lower layers. If n_l is the number of nodes in the overlay subscribing to receive layer l, then $n_l >= n_{l+1}$. A certain node i subscribing to receive some layers, will then need to distribute its upload capacity among these layers. In order to improve the average stream liveness across all the overlay nodes, the topology needs to prioritise the distribution of the most popular layers. This implies that nodes that are closer to the video source need to dedicate more upload capacity to the lower layers. This approach, however, may lead to quality bottlenecks, in which there is not enough capacity left to distribute the higher layers. To avoid quality bottlenecks, a node will need to first distribute one copy of the highest quality layers before it dedicates its residual capacity to distribute multiple copies of the lower quality layers.

A simple capacity allocation policy implementing these high-level objectives is the following:

- If $C_i \leq \sum_{l \in L_i} b_l$, where C_i is the upload capacity of node i, L_i is the set of layers received at node i, and b_l is the bitrate of layer l, then the node distributes one copy of the highest layers in L_i, until its upload capacity is exhausted.
- If $C_i > \sum_{l \in L_i} b_l$, then the node distributes one copy of all layers in L_i, and until its residual capacity is exhausted, it allocates b_l to layer l with probability proportional to the number of nodes subscribed to this layer across the overlay, i.e. $n_l / \sum n_l$.

Copyright © ENVISION Consortium, February 2012

4.1.3.1.2 Overlay Connection Selection

To determine the overlay connections for each peer, we will explore three different methods: *next-best-node*, *k-medoids* and *match-making*. Each of these methods is applied for each layer *l* separately, assuming the node upload capacity C_i^l as allocated by the previous step. To evaluate the outcomes of these algorithms we will use two metrics per participant node. The first one, the *stream liveness* $l(n_i) = \max_{l \in L_i}\left(l_l(n_i)\right)$, models the maximum delay from the source across all the layers received at a given node. The second one, the *stream quality* $\max_{l \in L_i}\left(l\right)$, models the highest quality layer that a peer downloads. An example topology that has been built for an overlay of 1000 nodes using one of the techniques discussed below is depicted in Figure 3.

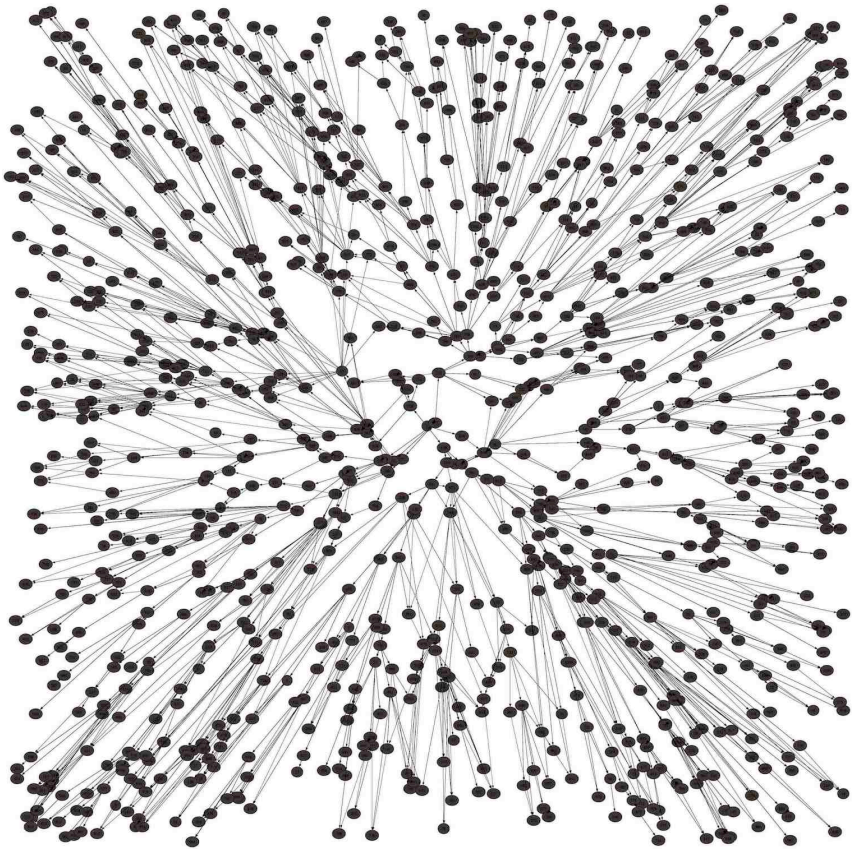

Figure 3: Example Overlay Topology

4.1.3.1.2.1 Next-best-node method

In *next-best-node*, the overlay topology is built by adding one node at a time, selected based on some optimisation criterion. The procedure is as follows:

1) $N_c = \{n_o\}$, $N_d = N - \{n_o\}$, where N_c is the set of connected nodes with enough available capacity for uploading the layer, n_o is the source node and N_d is the set of disconnected nodes
2) select n_i from N_d, with the maximum value for the optimisation metric v_o in set N_d; the following options are considered for v_o:
   - network propagation delay from the source $d(n_o,n_i)$
   - liveness $\min_{n_k \in N_c}\left(l(n_k) + d\left(n_k, n_i\right)\right)$, where $l_l(n_k)$ is the accumulated delay across all the overlay hops connecting node n_k to node n_o for layer *l*

Copyright © ENVISION Consortium, February 2012

- node power as capacity over delay from the source $C_i^l / d(n_o, n_i)$

3) select n_j from N_c, with the maximum value for the optimisation metric $v_n(n_i,n_j)$; the following options are considered for $v_n(n_i,n_j)$:
   - one-hop network propagation delay $d(n_j,n_i)$ and network propagation delay to the source $d(n_0,n_j)+d(n_j,n_i)$
   - liveness $l_l(n_j) + d(n_j, n_i)$
4) create a connection between n_i and n_j, move n_i from N_d to N_c, and remove n_j from N_c if it has exhausted its upload capacity
5) repeat from the 2nd step until N_d is empty

4.1.3.1.2.2 k-medoids method

In *k-medoids*, the *Partitioning Around Medoids* (PAM) algorithm [TK06] is used to recursively split the overlay to a number of clusters per iteration and select one node to distribute the stream per cluster. The advantage of this technique compared to the *next-best-node* is that it takes into account the position and the importance of the node compared to other nodes in its area. The *k* parameter is determined by the upload capacity of the node selected at each iteration. A recursive procedure for the selection of the medoid nodes is performed. In the first iteration *k* is calculated based on the capacity of the source node, i.e. $k = C_{n_o}^l / b_l$. The recursive procedure *find_kmedoids(n_p,k,N_d)* is as follows:

1) if $|N_d| \leq k$, create a connection between the parent node n_p and each node in N_d and return
2) select *k* nodes from N_d randomly to become the medoids $m_i \in M$
3) create a cluster $G_i = \{m_i\}$ for each medoid m_i, and assign each node n_j in $N_c \setminus M$ to the cluster G_i of the m_i with the minimum delay to the node, i.e. $d(m_i, n_j) = \min_{m_k \in M}(d(m_k, n_j))$
4) for each cluster G_i and each node n_j in G_i, do:
   a) swap m_i with n_j and calculate the cost for the new configuration as the average distance between all the nodes in G_i and the new medoid
   b) if the new cost is less than the new cost, i.e. if $\underset{n_k \in G_i}{avg}(d(n_j, n_k)) < \underset{n_k \in G_i}{avg}(d(m_i, n_k))$ then replace m_i with n_j
   c) proceed with the next node in G_i
5) repeat from the 2nd step until no medoid replacement is performed
6) for each m_i create an overlay connection between the parent node n_p and m_i, and run *find_kmedoids(m_i , $C_{m_i}^l / b_l$, G_i)*

4.1.3.1.2.3 match-making method

In *match-making*, a sender and a receiver selection policy is defined, and a match-making algorithm determines the most desirable outcome for all the nodes acting from these two perspectives. This last method is the most straightforward to directly apply in a distributed fashion, as the overlay construction policies are already expressed as independent sender and receiver policies. The asymmetry of information that may create discrepancies between the solutions found in each node, can be addressed by operating on smaller sets of nodes as determined by the distributed overlay coordination functions (see section 4.1.3.3). The match-making procedure is as follows:

Copyright © ENVISION Consortium, February 2012

1) Nc = {no}, Nd = N – {no}
2) for each node in ns in N and each node nr in Nd calculate the value the node nr represents as a receiver to node ns as a function rv(ns,nr); the following options are considered for rv(ns,nr):
   - one-hop network propagation delay $d(n_s,n_r)$
   - node power as capacity over one-hop delay $C_r^l / d(n_s,n_r)$
3) for each node in ns in N and each node nr in Nd calculate the value the node ns represents as a sender to node nr as a function sv(nr,ns), rank the nodes in N based on sv(nr,ns) into a propose list Pr for node nr and select the head of the list to be the next sender to propose for node nr, denoted by pr; the following options are considered for sv(nr,ns):
   - one-hop network propagation delay $d(n_s,n_r)$ and network propagation delay to the source $d(n_0,n_s)+d(n_s,n_r)$
   - node power as capacity over delay to the source $C_s^l / \left(d(n_o,n_s)+d(n_s,n_r)\right)$
4) select a node ni from the set of disconnected nodes Nd
5) if pi the next sender to propose for node ni has available capacity then:
   a) move n_i from N_d to R_{p_i}, the set of receivers of p_i, and update p_i's available capacity; else:
   b) if the value of n_i as a receiver is greater from the value of any of the existing receivers of p_i, i.e. if $rv(p_i,n_i) > \min_{n_j \in R_{p_i}} \left(rv(p_i,n_j)\right)$ then move the receiver with the minimum value in R_{p_i} back to N_d and move n_i from N_d to R_{p_i}; else:
   c) set p_i to the next node in the ranked list of senders for n_i
6) repeat from the 4th step until Nd is empty

4.1.3.2 Mobilisation of ISP and Participant Resources

The overlay topology construction methods described above do not take into account the existence of resources offered by the ISP and by the overlay participants (beyond their upload capacity for the distribution of the stream). The following sections elaborate on the possibilities for incorporating such resources in the topology construction framework described above.

4.1.3.2.1 ISP Multicast

By using ISP-provided multicast services, a large number of nodes can receive the stream or a subset of the layers of the stream, while only consuming the upload capacity of a single sender. The overlay topology construction algorithm can be enhanced to model this situation by also taking as input the subsets of nodes can use the multicast transmission mode among them.

As a first approach, the *next-best-node* and *k-medoids* policies described in section 4.1.3.1 can still be applied with the following modification: when a node is selected that belongs to a subset that can use multicast transmission, it automatically becomes the multicast source for this subset and all other nodes in the subset are added to the connected node sets. A modified *k-medoids* method will be investigated to select medoids in the clusters defined by the multicast node subsets.

Finally, we aim to investigate two distinct cases. In the first one, multicast resources can be used without any restrictions; in the second one, a cost is associated with the use of multicast transmission at any particular node subset for any particular stream layer. In this case, we will consider a given budget that can be spent by the overlay for the transmission of all layers to all the nodes.

Copyright © ENVISION Consortium, February 2012

4.1.3.2.2 ISP High-Capacity Nodes and Participant Relay Nodes

High-capacity nodes and participant relay nodes are two types of resources that perform an identical function for the overlay: download and replicate a stream or a part of the stream determined by the demand for this part of the stream across the overlay rather than by the requirement to download and consume the video stream.

The value of integrating relay nodes to the overlay topology comes from the fact that they can download part of the stream once and upload it many times. The higher the replication factor is, the more value the overlay gains from these relay nodes. In the case of ISP-provisioned high-capacity nodes, the finest granularity of the part of the stream to replicate will be a stream layer, as they have sufficient upload capacity to replicate a stream layer many times over. This is not true, however, for participant relay nodes. In this case, a fraction of a stream layer, e.g. all chunks with even identifier or divisible by three and so on, may be more appropriate.

While the overlay connection selection techniques described in section 4.1.3.1 can be used without significant modifications to incorporate relay nodes, a dedicated capacity allocation policy needs to be designed to determine which layers a relay node will subscribe to, in order to increase as much as possible the benefit it brings to the overlay. One such policy considered at this point is based on *scores*, and it aims at capturing the local demand for particular layers. On this basis, the algorithm then determines which layers to download and how much upload capacity to allocate to each of them. All consuming nodes in the overlay cast a vote for each layer at each relay node using as weight their delay to the relay node. Each relay node allocates its capacity to the highest scoring layers, proportionally to the sum of votes they have received, and only if they can achieve a minimum replication factor.

Similarly to the multicast case, we aim to investigate one case with no constraints and one case with a cost associated with the use of participant and ISP relay nodes and a given total budget that can be spent by the overlay.

4.1.3.3 Distributed Overlay Coordination

The objective of the work in the previous sections is to find good solutions using complete information for the overlay, therefore setting a benchmark for the evaluation of more distributed approaches. This section aims at investigating the impact of restricting the information available to the overlay optimisation algorithms to their performance. The algorithms described in section 4.1.3.1 will be modified to operate on groups of nodes that will be created based on the ISP domain and possibly the PID they belong, spanning a range of sizes. One such example clustered overlay topology for 1000 nodes can be seen in Figure 4.

Dedicated policies will be investigated to ensure that there is sufficient overlap between these groups or that the functions operating at each group have sufficient information about nodes in other groups, to ensure good connectivity across the different groups. The specification of these policies is left for future work.

Copyright © ENVISION Consortium, February 2012

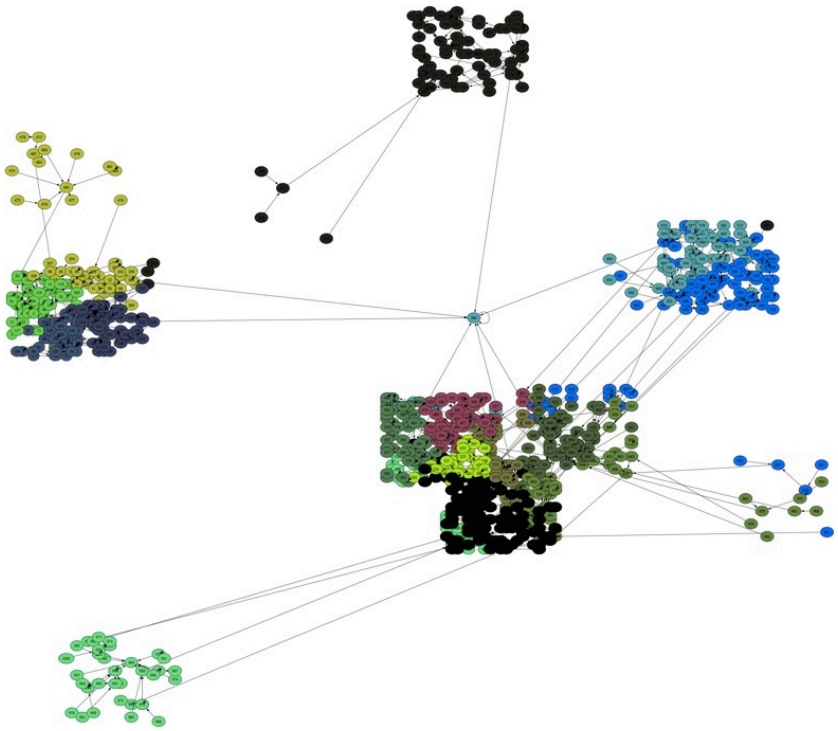

Figure 4: Example Clustered Overlay Topology

4.1.3.4 Interface with the Chunk Selection Policy

In ENVISION, a pull-based algorithm that relies on receiving nodes requesting data chunks from a set of sender nodes is being developed as part of the work in WP5. The work described in this section will result in a joint prototype combining the overlay topology construction policies developed in WP4 with the chunk selection policies developed in WP5. A sender selection function at each receiving node will be responsible for gathering information about candidate senders and competing receivers and for selecting the best senders for the node based on this information and given the target set of quality layers determined by the smoothing functions in WP5. The chunk prioritisation and request scheduling functions in WP5 will use the list of candidate senders selected by the overlay topology construction, in order to issue chunk requests based on their chunk prioritisation logic.

4.2 Interactive Video Content Distribution

4.2.1 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 minimize the end-to-end latency
- The system must be able to integrate network services into the overlay topology and optimize the topology accordingly

Copyright © ENVISION Consortium, February 2012

4.2.2 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 discuss an optimization procedure that enables the system to decide where and when to invoke CINA enabled network services, such as the High Capacity Node.

Section 4.2.3.1 therefore first discusses the implementation of the system. In particular it details the respective software architecture components and their relationship to each other, followed by a discussion of the various topology strategies implemented by the Tree Manager module in section 0. Finally in section 4.2.3.3 we discuss the Distribution Tree Optimisation procedure, which allows the overlay to decide where to integrate a service like the High-Capacity Node Service. For this procedure we develop a network model and problem formulation and give first insights into the approach we are going to investigate. We close the chapter by a short conclusion giving an outlook on the next steps as well as the demonstration and evaluation scenarios.

4.2.3 Specifications

The IVCD layer is divided into two building blocks: (i) The tree manager, which maintains a view on the current network and (ii) the streaming peer which forwards and processes the media content. Nodes are not only receiving content as a child node but may also in parallel send content as parent node. Therefore the nodes create a structured peer-to-peer overlay tree topology, controlled by the tree manager.

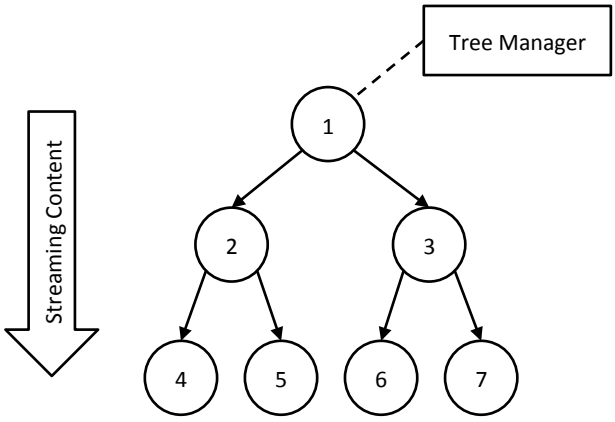

Figure 5: Tree Overview

Figure 5 illustrates the tree manager attached to the root node of the tree. Each source node of a streaming tree spawns a tree manager instance for each content id the node provides. Each node can be classified by a number of parameters like available bandwidth or processing power, as well as delay to other participants already in the tree. When new nodes are joining the overlay, they contact the tree manager. The tree manager then chooses the best contact node (parent) for the joining node and sends the contact information. The tree manager thus controls the overlay topology.

Copyright © ENVISION Consortium, February 2012

4.2.3.1 Software Architecture

In the following section a brief overview of the software architecture is given. First the overall system is illustrated, then each module is described in more details.

A streaming tree node consists of several threaded modules outlined in Figure 6. Every node has one StreamAcceptor that is responsible for stream receipt. It passes incoming data blocks to the PeerControl unit, where they are put into a buffer queue and supplied to the StreamSubmitters. Additionally, incoming data is copied into a buffer delivered by the Application, if existent, to provide media playback. PeerControl manages the node's information about its environment. It stores known peers and knows about parent and child nodes. For each of its children, the node creates a StreamSubmitter instance, which accesses stream data blocks and send them to their related child node. Heartbeat and SignalingSystem are support modules.

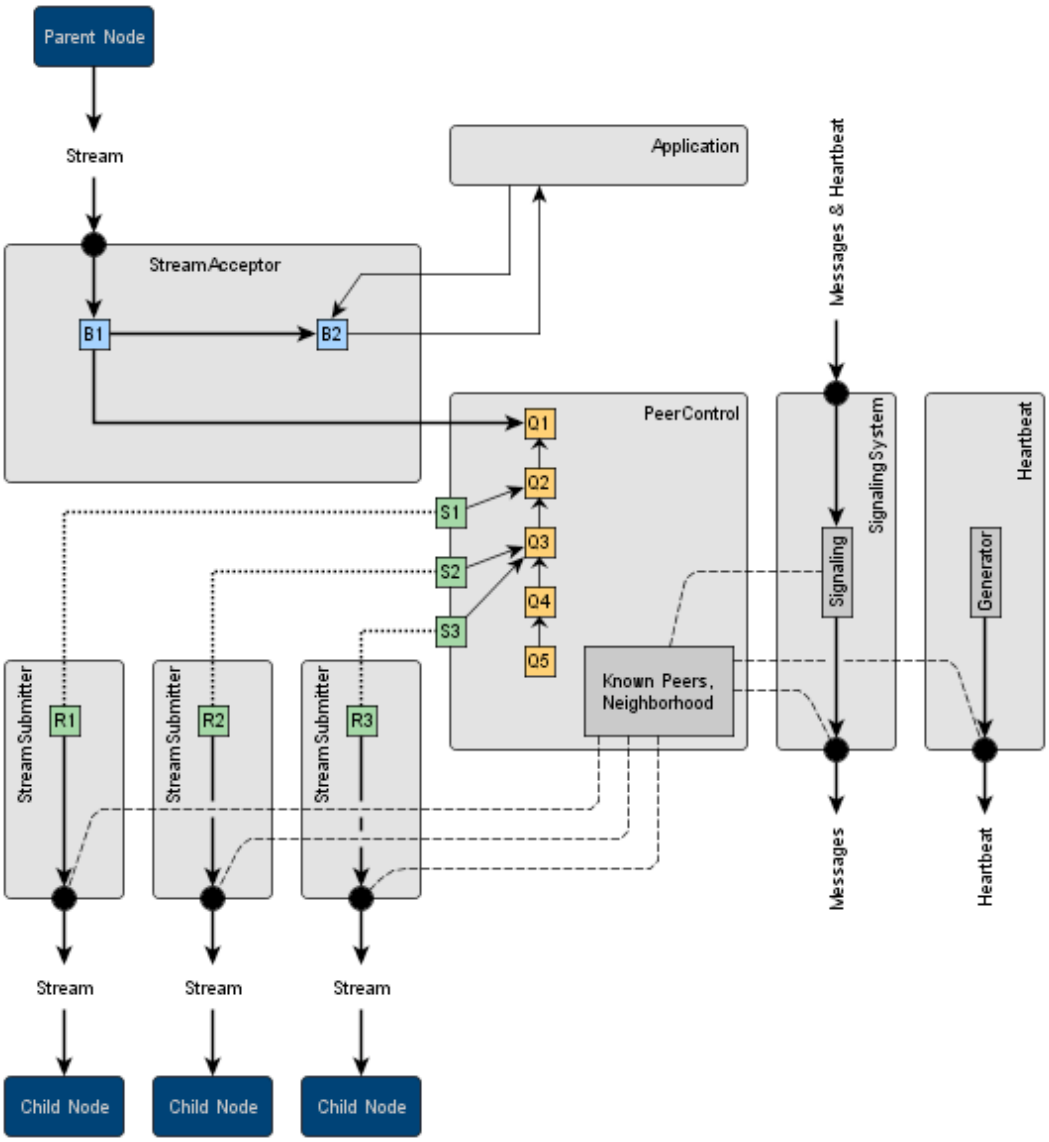

Figure 6: Node architecture overview

In Figure 6, dark blue rectangles visualize other nodes, such as the parent node and (exemplary) three child nodes. Rectangles filled light gray are the node's internal components. Black circles placed on their boundaries indicate ports that are used to receive or send data or messages. Blue squares captioned B1 and B2 are data block buffers, orange ones (Q1 – Q5) are elements of the

Copyright © ENVISION Consortium, February 2012

stream buffer queue. S1 – S3 are stub objects pointing at a single queue element each. R1 – R3 are references to those stubs, using a 1:1 relationship, which is visualized by dotted lines. Dashed lines indicate a dependency of the node's neighborhood information, managed from within PeerControl unit. They control target socket addresses, for example. Black arrows demonstrate data flow.

PeerControl is the central instance of a node and holds references to the other modules SignalingSystem, HeartBeat, StreamAcceptor and StreamSubmitter. These modules are instantiated on creation of a PeerControl object. Each of them runs in its own lightweight thread and acts autonomously.

4.2.3.1.1 PeerControl

The PeerControl unit shown in Figure 7 is the central instance of every streaming tree node. It holds and manages data structures of global interest. Information about known peers, the node's neighborhood and the stream buffer queue are provided to other modules of the node architecture via interfaces at PeerControl.

The SignalingSystem is an active unit which can receive and send messages from resp. to other nodes. The Heartbeat contains a generator creating heartbeat messages frequently. Both modules are described more detailed below.

All information concerning a node's environment is stored in the neighborhood structure. This unit defines target addresses and ports for messaging and stream submission. On alteration of the neighborhood, all affected target addresses change implicitly as well. Dashed lines in Figure 6 and Figure 7 indicate dependencies of information saved in the neighborhood structure. For example, every message that is sent, obtained the target address from the neighborhood.

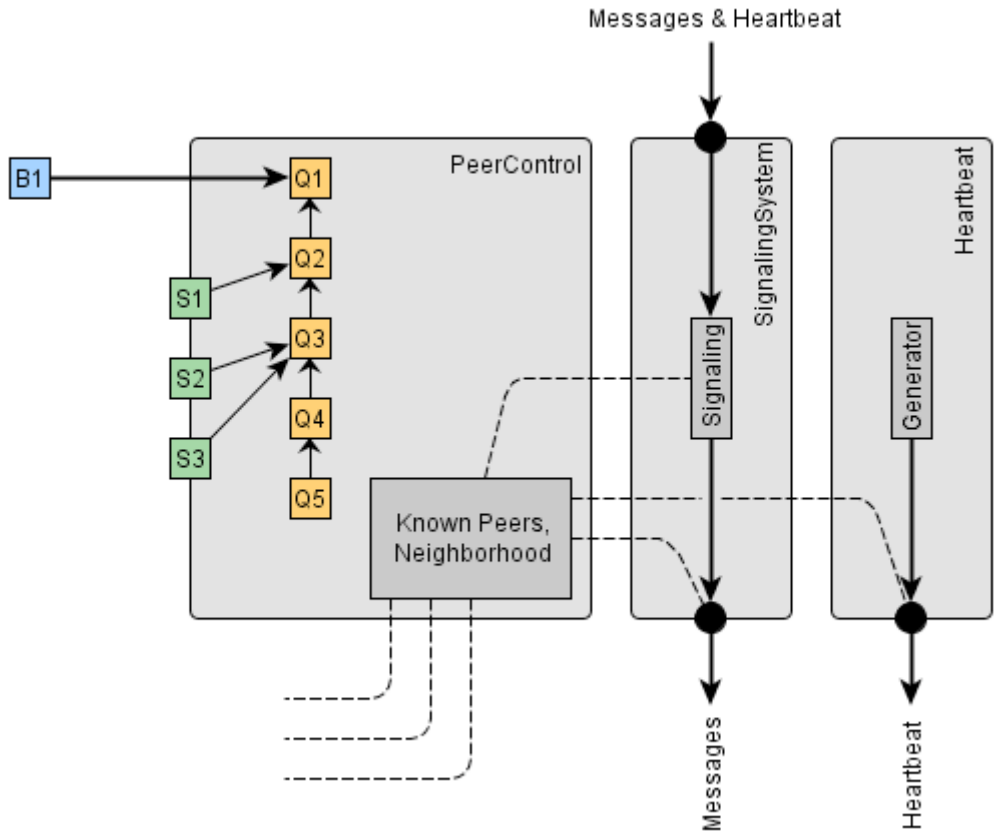

Figure 7: PeerControl module

Copyright © ENVISION Consortium, February 2012

The stream buffer queue visualized by the orange squares Q1 – Q5 holds stream data blocks for retransmission. Blocks are inserted when the StreamAcceptor passes a new stream data block B1 to PeerControl. Each newly received block is pushed to the head of the queue. The stubs S1 – S3 are related to the node's StreamSubmitter instances and each point to one data block within the queue. When the certain StreamSubmitter has finished transmission of the current block, the pointer proceeds one block towards the queue's head. The stream buffer queue is bound to a dedicated garbage collection that deletes blocks from the queue's tail, which are not referenced by any stub.

4.2.3.1.1.1 Heartbeat

The Heartbeat module generates a frequent heartbeat using a tree-wide defined interval. For each beat, the PeerControl module is used to create heartbeat messages by the SignalingSystem. Heartbeat messages are sent to all child nodes and the parent, if existent. The heartbeat module is not responsible for monitoring of incoming heartbeat messages from parent or child nodes. Heartbeats arrive – as every node-interaction message does – at the node's signaling port, which is located in the SignalingSystem.

4.2.3.1.1.2 SignalingSystem

The SignalingSystem is an instance which is capable of creating and sending various signal messages to other nodes, as well as receiving and parsing messages from other nodes. This subsystem is used for every interaction between nodes, such as join requests or recovery information exchange for example. The SignalingSystem handles messages concerning the node's streaming tree position autonomously, such as the complete tree joining procedure. It requests an ID at the TreeManager. On receipt, it passes the ID to the PeerControl module and requests the Tree Manager for a tree position. If the position assignment message is received, it requests the delivered contact node for a child join. On acknowledgement, the SignalingSystem invokes the PeerControl module to set the parent node. Messages that are not related to the node's position in the streaming tree, namely incoming heartbeats or messages concerning pre-failure network exchange, trigger a notification of the PeerControl unit, which has to handle them.

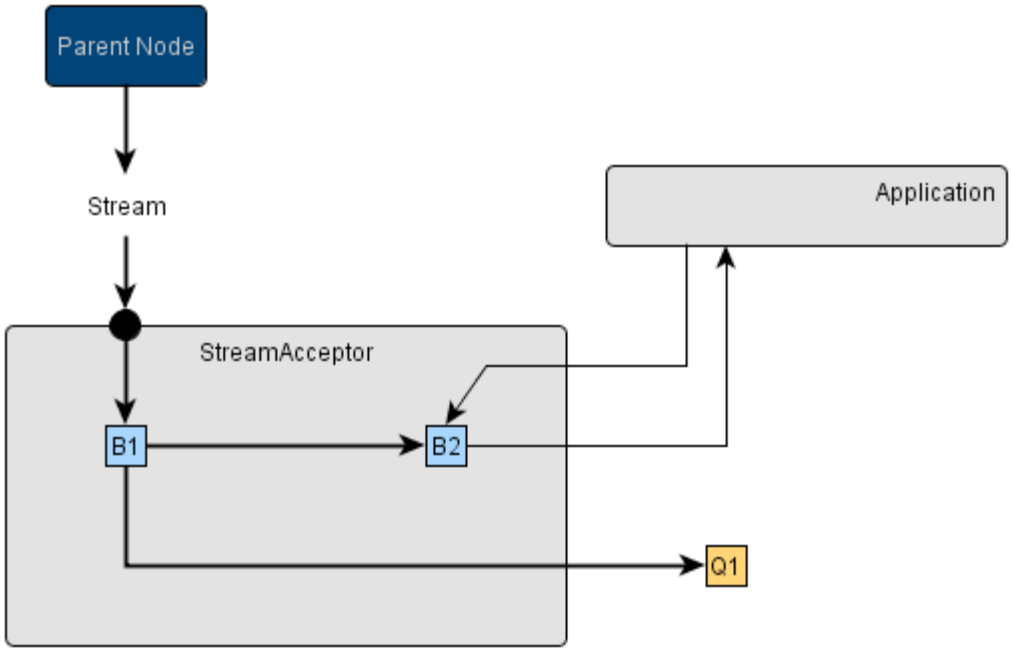

Figure 8: StreamAcceptor module

Copyright © ENVISION Consortium, February 2012

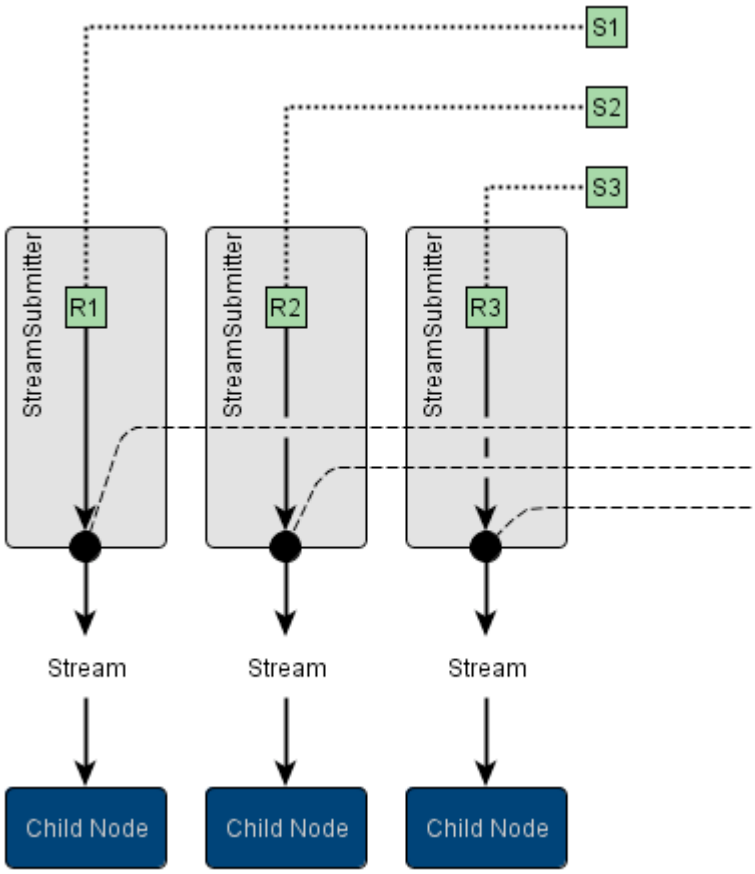

Figure 9: StreamSubmitter module

4.2.3.1.2 StreamAcceptor

The StreamAcceptor module (Figure 8) receives the stream from the parent node. It stores incoming data blocks in a buffer B1 and passes them to the PeerControl unit as fast as possible to be capable of receiving eventually following blocks. Furthermore, the StreamAcceptor delivers received data to the application layer above. For this purpose, the buffer is copied into another data block provided by the application.

4.2.3.1.3 StreamSubmitter

Each StreamSubmitter (Figure 9) directly reads blocks from the stream buffer queue stored in PeerControl and sends them to the related child node. A node has as many StreamSubmitters as it has child nodes to be supplied. While certain output ports are managed autonomously, the target addresses and ports are obtained from the neighborhood stored in PeerControl.

To always guarantee fast transmission, StreamSubmitters read directly from data blocks from within the queue. These blocks are referenced by the stubs S1 – S3, which are related to one StreamSubmitter each. When a submitter has finished transmission of a block, it invokes the stub to switch over to the next block immediately. If the queue head is reached, the StreamSubmitter stops transmission until a new block is enqueued.

4.2.3.1.4 Stream Buffer Queue and Garbage Collection

Figure 10 provides a detailed view of a stream buffer at runtime. The large orange blocks in vertical alignment represent data blocks of the media stream. They are arranged in a linked list, each pointing to its successor, so the oldest block is that one indicated by the tail-pointer. The head pointer links to an empty data block on top of the queue. When new data has to be enqueued, it is filled into the empty head-block and a new empty block is added in advance.

Copyright © ENVISION Consortium, February 2012

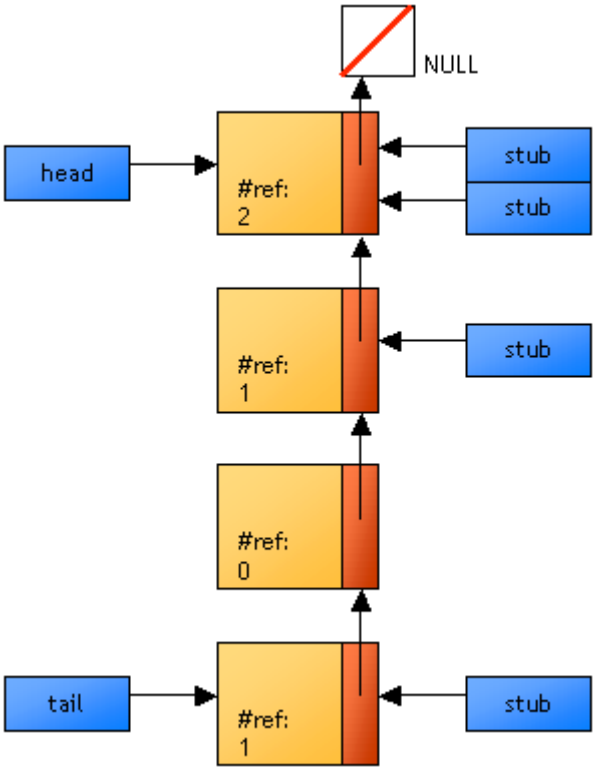

Figure 10: StreamQueue example

The stubs to the right side of the visualization each point to one data block within the queue. Each of them is bound to one StreamSubmitter and indicates the certain block, which is currently being transmitted. When the submitter has finished sending the certain block, the stub pointer advances one block in the stream buffer queue unless it has reached the queue's head. Every time a stub changes from one data block to another, it has to unregister its reference at the old block and register itself at the new one. The amount of registered references is shaped out as #ref in each data block in Figure 10. Once the queue's head is reached, the stub pointers do not advance until new data has been inserted.

To prevent the queue from growing to infinite length, a dedicated garbage collection frequently starts at the queue's tail and deletes all blocks that have no registered references. Additionally the garbage collection checks the distribution of stub pointers over the whole queue and if a certain queue length has been exceeded. If it recognizes a stub pointer to be stuck at the queue's tail, for example because message loss has occurred, it may perform a hard reference shift and redirect the pointer to the queue's head. The affected StreamSubmitter then has to restart transmission of the new block and continues sending media data.

4.2.3.2 Tree Manager

As discussed before the Tree Manager is the functional entity that has the control of the streaming tree topology. Figure 11 shows the message exchange of the Join procedure when a new node wants to join the tree. In particular the peer sends a JoinRequest message to the Tree Manager. The initial contact point thereby has to be specified by the application; there is no discovery procedure implemented in the content distribution layer. The Tree Manager then calculates the peer ID and the position of the new peer in the tree. Different strategies for this calculation are available, such as assignment of a random free spot, using longest prefix match to assign neighbours and finally incorporating CINA network information into the calculation. A more detailed discussion of the three strategies is given below.

Copyright © ENVISION Consortium, February 2012

After the joining peer has received its ID and the contact information of its new parent peer it sends a handshake message to the parent peer. If the parent peer acknowledges the handshake the joining peer finished the joining procedure by reporting to the Tree Manager its current position and the number of peers it is willing to serve (fanout).

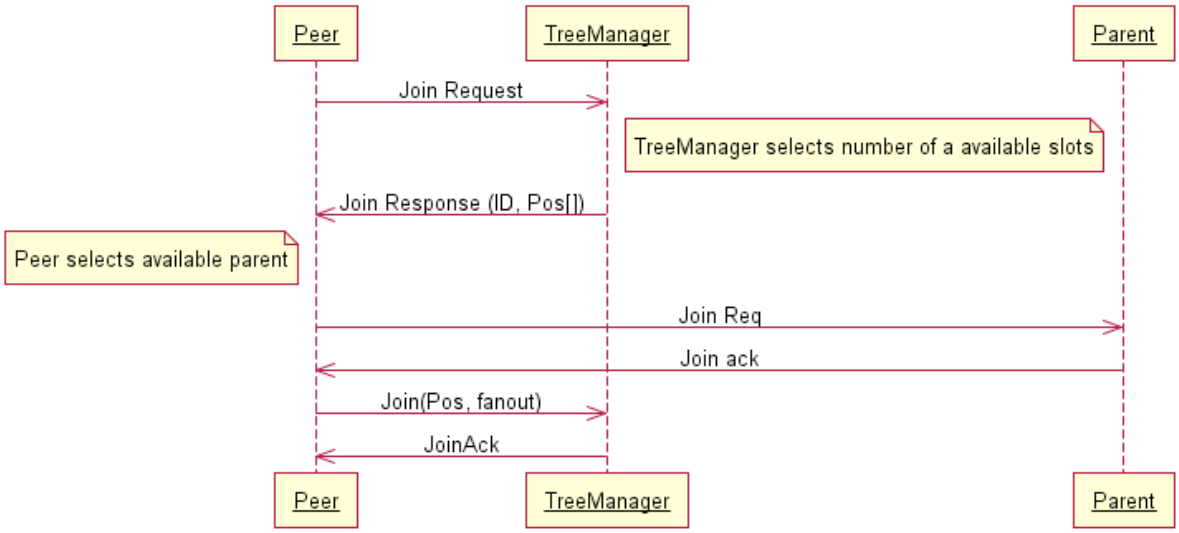

Figure 11: IVCD Join Procedure

4.2.3.2.1.1 Random Topology

In the random topology strategy the Tree Manager selects a free slot for the joining peer in the level with shortest distance to the source. If there are multiple slots on this level available, one is chosen randomly. As all peers that are already integrated into the topology have reported the amount of peers that they are willing to serve, the Tree Manager can make sure that it does not assign the joining peer and on the same time overloads a branch of the tree. This strategy typically results in a topology where the underlying network topology and the overlay topology do not match. Thus content items might be sent over unnecessary long distances.

4.2.3.2.1.2 Longest Prefix Match Topology

In the longest prefix match topology strategy the Tree Manager compares the IP addresses of the joining peer and those peers that have a free slot and calculates the network distance between them. It then returns the peer which is closest to the joining peer. This strategy typically results in trees that have a closer match between network topology and overlay topology, in particular if several peers from the same subnet participate in the same session, as for example in typical videoconference scenarios. However due to the fragmentation of the internet having a long IP prefix in common does not automatically mean that peers are close to each other in the network. In addition there is a tradeoff between assigning a peer to its closest neighbor w.r.t. the IP prefix and the balance of the tree topology.

4.2.3.2.1.3 CINA Enabled Topology

In the CINA enable topology strategy the Tree Manager uses the CINA client library to perform CINA queries to its local CINA server. It retrieves network information in the form of cost maps to get a hint from the network about where to best assign this peer. This network information is then taken into account when choosing the free slot for the joining peer. As this strategy incorporates explicit recommendations from the network the overlay topology typically has a higher overlap with the demand of the network than the aforementioned strategies. Additionally the topology can be adjusted to the requirements of the application, for example by retrieving cost maps about the geographical distance or about typical latency between network regions.

Copyright © ENVISION Consortium, February 2012

4.2.3.3 Distribution Tree Optimization

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

4.2.4 Conclusion

In this chapter we have detailed the software implementation of the Interactive Video Content Distribution system. We have put the focus on the enhancements of a pure overlay approach brought by CINA enabled network information and services.

The next steps will be the finalization and integration of all software components needed for the complete demonstration scenario as detailed in D6.1. This demonstration scenario will show the use of network information and services by the IVCD system through the CINA protocol, including the implementation of the High Capacity Node, the CINA Client and Server libraries and data gathered by the Network Monitoring modules.

4.3 Caching Optimisation based on Social Network Data

This section has been suppressed from the public version of this deliverable. A full description of the Caching Optimisation based on Social Network Data work is available at [THT+12].

4.4 CDN Node Selection Optimisation with CINA Routing Costs and Dynamic Overlay Monitoring

4.4.1 Problem Statement

In this section we are exploring the use of CINA costs calculated using BGP information combined with overlay measurements in order to optimise the content distribution for a CDN overlay. Knowing how costly is a connection between two IP addresses in real time is crucial for a CDN. It enables the CDN to take optimum decisions not only for the selection of the delivery node for an end user, but also for the distribution of the content between the CDN nodes.

The CDN optimisation process requires information about the network performance for any Internet destination. It needs also to be aware of dynamic network conditions such as latency, congestion, packet loss, etc. However, obtaining accurate information for the network topology and the traffic conditions for the entire Internet is very difficult, due to the extremely large amount of network elements and providers and also because of the diversity and complexity of the different network management processes involved.

4.4.2 Approach

BGP is today the only protocol that can provide information about the end-to-end path to any destination that is available to an ISP, and it can be used to provide a rough estimation of the performance along the path. This information can be used by the CDN overlay optimisation functions to select appropriate delivery nodes based on their proximity to end users in destinations that are not previously known to the CDN. As the CDN applications do not have direct access to up-to-date BGP information, they rely to the CINA interface for obtaining costs calculated by the ISPs that reflect the BGP hop count.

This process works well as a first approximation of the network performance but has several drawbacks:

- BGP provides coarse grained information about network paths. Its smallest unit is the autonomous system which generally comprises a large network with a lot of routers and links. BGP routes cannot be used to estimate the number of links that the traffic traverses inside the autonomous systems.

Copyright © ENVISION Consortium, February 2012

- BGP does not carry any information related with the dynamic network conditions, which, in the presence of congestion, may severely impact the network performance metrics including latency, throughput, etc.

Therefore, the network performance estimation obtained through BGP needs to be enhanced with monitoring information obtained at the application layer. Analysing incoming connections at the CDN nodes makes it possible to know some important network metrics such as round trip times, packet loss or jitter between delivery sites and final users. This type of passive monitoring gets end-to-end measurements seeing the network as a black box.

As loss is typically low, latency is most frequently the factor with the greatest impact in the performance of TCP connections. Therefore, CDN node selection should focus mainly in redirecting users to nodes with the lowest latency. Moreover, as latency varies over the time depending on network conditions such as congestion, suboptimal routing or partial failures, a continuous latency measurement is needed to keep the latency map between CDN nodes and final users up to date.

The method we are going to use is to measure the round trip times of incoming connections at the initial TCP three-way handshake. Examining the time between SYN-ACK and ACK messages of each incoming connection, gives a good estimation of average RTT between the two endpoints of the connection as it is stated by [SM06].

With this technique, we can detect node selections that are suboptimal, when for example high RTT (> 150 ms) values are observed repeatedly for end user IP addresses coming from a certain origin network. New end users from the same network will be subsequently redirected to different CDN nodes.

This method can be used to refine the decisions that are taken based on the CINA costs reflecting the BGP routing information and at the same time it is very cheap to implement in resource consumption. It only focuses in networks with a significant amount of CDN end users so as not to waste resources in monitoring other networks.

The detailed specifications for this approach can be found in the following sections. The proposed system will be evaluated with the development of a proof-of-concept prototype and its deployment in TID's testbed.

4.4.3 Specifications

4.4.3.1 CDN Architecture

In the next picture you can see a general view of the CDN components and how they interact with each other in a typical use case scenario.

4.4.3.2 Component Description

- **CDN Nodes.** These are the content delivery nodes. The large amount of servers where the end users connect to for content downloading
- **Origin Servers.** These are the entry point servers for CDN customers. They are the primary location of content in the CDN. Once the content is there it is available for delivery by the CDN nodes.
- **Tracker.** The main controller of the CDN. It continuously receives snapshots of the CDN nodes resource usage statistics. With these data plus the CINA costs, the tracker can decide which is the best node to serve a specific end user, both in terms of network and in terms of application resources.

Copyright © ENVISION Consortium, February 2012

- **Network topology server.** It is providing the CINA cost for an overlay connection between a pair of Internet endpoints. It is one of the information sources used by the CDN node selection algorithm in the tracker.
- **DNS.** The authoritative DNS server for the CDN domain. It makes the resolution of CDN host names, delegating the CDN node selection for a specific user to the tracker.

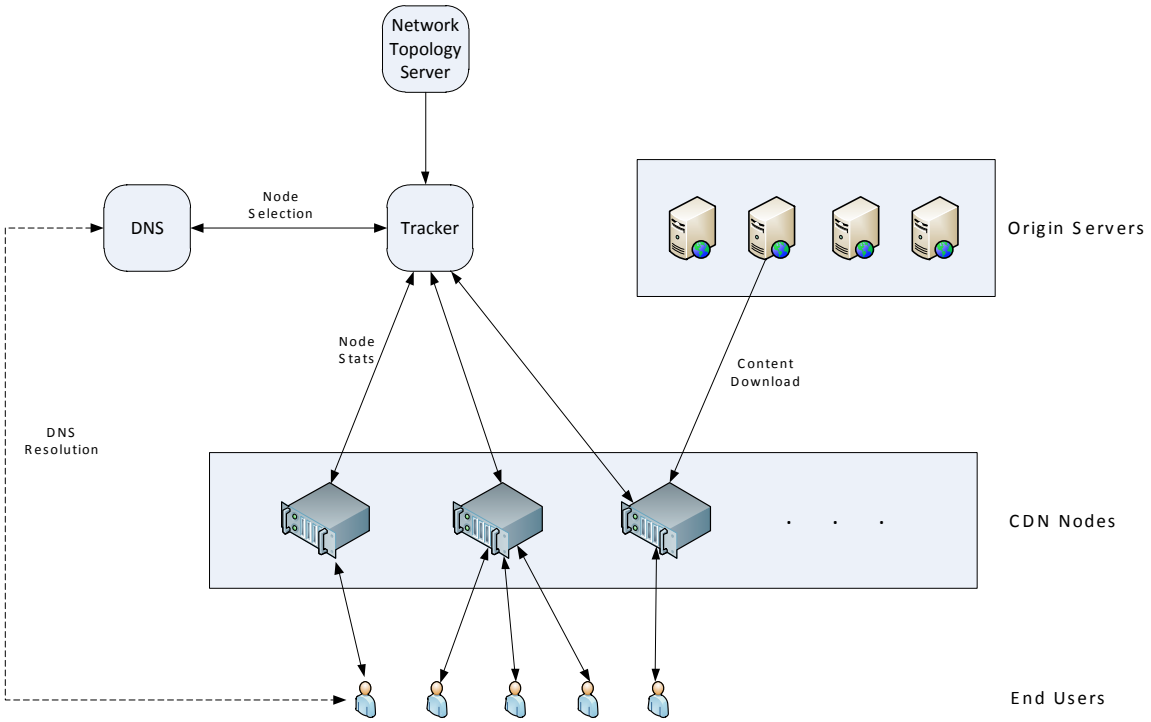

Figure 12: CDN Architecture

4.4.3.3 Use Case Scenario

When an end user is going to download a video hosted in the CDN, she uses a URL with the CDN domain name which needs to be resolved by a DNS server. Node selection is made transparently to the end user in the DNS name resolution process, and it involves the following steps:

1) The end user DNS server requests to CDN DNS server to resolve a hostname in its domain.
2) The CDN DNS asks the tracker for a subset of the CDN nodes suitable for delivering the content to this end user.
3) The tracker selects a couple of nodes taking into account content distribution policies, performance and load metrics, as well as the CINA costs corresponding to BGP routing data.
4) The tracker returns these IP addresses to the CDN DNS server who uses them to build the reply to the end user DNS server.
5) Finally, the end user selects one of these addresses, connects to it and requests the video.

4.4.3.4 Node Selection Process

Node selection is made on the tracker based on two inputs, the specific content the user is trying to consume and the IP address of the end user's DNS server. The first input is used to make the selection optimum in terms of content distribution across the CDN whereas the second one is used to make the best selection in terms of network paths.

The process starts with the host name resolution of a CDN URL. The CDN host names have the following pattern: B3.cdn.telefonica.com, where the number identifies a specific customer container,

Copyright © ENVISION Consortium, February 2012

which means that the CDN node selection is also aware of the content the user is going to consume, in order to make the content distribution efficient and optimize the cache hit ratio. As we have seen above, the tracker does the node selection taking into account load state of the nodes, content distribution and network criteria. For the network phase it uses the CINA costs corresponding to the inter-domain routing information.

This information is composed of two parts corresponding to the CINA network and cost maps:

- **The network prefix partitions set**. The IP address space is divided into a set of partitions or subsets according to the locations of the routers where those prefixes advertisements are received by the BGP protocol. So for example, all the prefixes advertisements received in the BGP routers located at Equinix IXP in London end up in the same partition.
- **The cost matrix**. It is a NxN matrix, where N is the number of partitions and the *(i,j)* entry represents the cost of transmitting data from any endpoint in the partition *i* to any endpoint in the partition *j*. That cost is calculated statically by the network operators combining some metrics, such as distance, number of hops, capacity of the links, etc.

These two data sets are used by the tracker to select the nodes following these steps:

1) Takes the end user's DNS IP address and locates its partition (P_i) matching the most specific network prefix in the partitions set.
2) Takes the subset of the CDN nodes, previously filtered based on content and load criteria and order them using the cost from P_i to their partitions using the cost matrix. Once the node list is ordered, the IP addresses of the first couple of nodes of the list are the tracker's response to the DNS.
3) The CDN DNS composes the answer with these addresses returned by the tracker.

4.5 ISP Resource Invocation with Cost Predictability

While some overlay applications may require a minimum level of QoS, others may be able to operate without hard QoS guarantees. A typical example is an interactive video application that cannot tolerate delays greater than 200ms, as opposed to live video applications which will respond to increased delay by increasing the viewing delay for their consumers. The invocation of ISP resources through the CINA interface will be in most cases associated with a particular pay per use pricing scheme. Those applications that do not require hard QoS guarantees may be willing to settle for a more relaxed statistical QoS in exchange for a predictable flat charge.

To enable ISP resource invocation with cost predictability, ENVISION proposes Temporal Rate Limiting (TRL), a purchase policy that permits overlay applications to allocate optimally a predefined purchase budget over a predefined period of time. Overlay applications can implement auto-scaling purchase policies by leasing (e.g., hourly) necessary amounts of resources like multicast transmission and high-capacity nodes (see [D3.2]) to satisfy a desired QoS threshold under their current demand and number of overlay participant nodes. The benefits of TRL have been studied and quantified analytically in the context of instantiating cloud resources in [OSL12]. A TRL pilot has been deployed on Amazon EC2 and a live validation has been performed in the context of a "blacklisting" application for Twitter.

5. CONCLUSION

This document elaborates on the functionality that is required to enable high-volume future media applications to be distributed over large and dynamic overlay networks operating in collaboration with the underlying ISPs.

Several possibilities have been identified regarding the collaboration between overlay applications and ISPs with the purpose of improving the overlay network performance modelling functions, either

Copyright © ENVISION Consortium, February 2012

by increasing their accuracy or by reducing their load. Although each ISP has visibility of the network conditions in its own domain, it is also in an advantageous position to operate observation points and collect information about the end-to-end performance experienced by traffic originating or terminating at its domain. Because of the confidentiality issues involved in directly exposing the value of network performance metrics, some of the identified options may only be reserved for CINA clients with privileged access to ISP information.

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 technique 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. Although sophisticated techniques are explored for determining an optimum clustering hierarchy based on network delay, the analysis of the n-casting protocol does not depend on their outcome. Rather, a clustering hierarchy based on geolocation that has been shown to have strong correlation with delay is adopted as a working assumption, decoupling thus these two research threads.

Overlay topology construction heuristics are studied in the context of live video streaming overlay applications, supporting heterogeneous devices with the use of SVC encoding. Special consideration is taken to avoid quality bottleneck in the overlay, by ensuring that the highest layers that are also more rare are distributed with priority at those nodes that are receiving them. These techniques are then enhanced to take into account the availability of multicast transmission capabilities, participant and ISP upload capacity resources. To allow for large scale implementations of these techniques, distributed heuristics are considered.

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 system involves a TreeManager function responsible for managing the overlay topology and a signalling protocol for the coordination between the participant nodes. The use of high-capacity nodes offered by the ISP at particular locations is investigated, with the formulation of the overlay topology optimisation problem and the description of a heuristic for creating good approximations.

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.

Copyright © ENVISION Consortium, February 2012

Finally, a proof-of-concept prototype is designed, aiming at evaluating the practical application of combining overlay monitoring information together with the CINA costs in order to improve the operation of a CDN overlay network. The CDN node that is closer in terms of network delay to any given user is first determined based on CINA costs calculated with BGP routing information, and refined later through the use of passive measurements from existing connections between CDN nodes and users of the same network location.

While some of these specifications are finalised, others are in an intermediate stage and will be further refined in the following reporting period. The evaluation specifications and preliminary results corresponding to the specifications described in this document can be found in [D6.1].

Copyright © ENVISION Consortium, February 2012

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.

[AGLO09] Sharad Agarwal and Jacob R. Lorch. 2009. Matchmaking for online games and other latency-sensitive P2P systems. SIGCOMM Comput. Commun. Rev. 39, 4 (August 2009), 315-326.

[AL09] Sharad Agarwal and Jacob R. Lorch. Matchmaking for online games and other latency-sensitive p2p systems. SIGCOMM Comput. Commun. Rev., 39(4):315–326, October 2009.

[Arr53] K. J Arrow. Social Choice and Individual Values. Cowles Foundation Monographs; New York: Wiley 1964, 1953.

[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.

[Bkc01] Andre Broido and kc claffy. Analysis of routeviews bgp data: policy atoms. In Network Resource Data Management Workshop, Santa Barbara, CA, May 2001.

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

[BOLO93] Jean-Chrysotome Bolot. 1993. End-to-end packet delay and loss behavior in the internet. SIGCOMM Comput. Commun. Rev. 23, 4 (October 1993), 289-298.

[BRCA09] Fabrìcio Benevenuto, Tiago Rodrigues, Meeyoung Cha, and Virgìlio Almeida. Characterizing User Behavior in Online Social Networks. In IMC, 2009.

[bur] Burstable Billing. http://en.wikipedia.org/wiki/Burstable_billing.

[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.

Copyright © ENVISION Consortium, February 2012

[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

[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

[D5.2] ENVISION deliverable D5.2, Refined Specification of Metadata Management, Dynamic Content Generation and Adaptation, Adaptation and Caching Node 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.

[EYR11] Vijay Erramilli, Xiaoyuan Yang, and Pablo Rodriguez. Explore what-if scenarios with SONG: Social Network Write Generator. http://arxiv.org/abs/1102.0699, 2011.

[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.

[FLM06] Michael J. Freedman, Karthik Lakshminarayanan, and David Mazières, OASIS: Anycast for Any Service, Proc. 3rd USENIX/ACM Symposium on Networked Systems Design and Implementation, (NSDI '06) San Jose, CA, May 2006.

[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.

[GWH07] Scott A. Golder, Dennis M. Wilkinson, and Bernardo A. Huberman. Rhythms of Social Interaction: Messaging Within a Massive Online Network. In C&T, 2007.

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

[hig] highscalability.com. YouTube Architecture. http://highscalability.com/youtube-architecture.

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

[IPPM-SC16] A. Morton, E. Stephan, Spatial Composition of Metrics, http://tools.ietf.org/id/draft-ietf-ippm-spatial-composition-16.txt, August 2010

Copyright © ENVISION Consortium, February 2012

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

[KAPA91] Phil Karn and Craig Partridge. 1991. Improving round-trip time estimates in reliable transport protocols. ACM Trans. Comput. Syst. 9, 4 (November 1991), 364-373.

[Ken] Niall Kennedy. Facebook's Photo Storage Rewrite. http://www.niallkennedy.com/blog/2009/04/facebook-haystack.html.

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

[KLPM10] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is Twitter, a Social Network or a News Media? In WWW, 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.

[LL08] D. Leonard, and D. Loguinov, "Turbo King: Framework for Large-Scale Internet Delay Measurements," IEEE INFOCOM, Apr. 2008.

[LLS07] C. Lumezanu, D. Levin, and N. Spring. Peerwise discovery and negotiation of faster paths. In Proc. Workshop on Hot Topics in Networks (HotNets), 2007.

[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] Raul Landa, Eleni Mykoniati, David Griffin, Miguel Rio, Nico Schwan, Ivica Rimac, Overlay Consolidation of ISP-Provided Preferences, to appear in the proceedings of the International Workshop on Cross-Stratum Optimization for Cloud Computing and Distributed Networked Applications.

[LNNK+05] David Liben-Nowell, Jasmine Novak, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. Geographic Routing in Social Networks. Proceedings of the National Academy of Sciences, 102:11623–11628, 2005.

[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.

[MAK06] Harsha V. Madhyastha, Thomas Anderson, Arvind Krishnamurthy, Neil Spring, and Arun Venkataramani. A Structural Approach to Latency Prediction. In ACM Conference on Internet Measurement (IMC), 2006.

[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.

[MIP+06] Harsha V. Madhyastha, Tomas Isdal, Michael Piatek, Colin Dixon, Thomas Anderson, Arvind Krishnamurthy, and Arun Venkataramani. iPlane: An Information Plane for Distributed Services. In Proc. of ACM OSDI, pages 367-380, 2006.

Copyright © ENVISION Consortium, February 2012

[MM02] P. Maymounkov, D. Mazieres, Kademlia: A Peer-to-peer Information System Based on the XOR Metric, IPTPS 2002.

[MRL+09] Daniel S. Menasche, Antonio A.A. Rocha, Bin Li, Don Towsley, and Arun Venkataramani. Content Availability and Bundling in Swarming Systems. In CoNEXT, 2009.

[MRS2002] N. M. Malouch, Z. Liu, D. Rubenstein, and S. Sahu. „A Graph Theoretic Approach to Bounding Delay in Proxy-Assisted, End-System Multicast", Tenth International Workshop on Quality of Service (IWQoS 2002)

[NRTV07] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani. Algorithmic Game Theory. Cambridge University Press, New York, NY, USA, 2007.

[NTW09] N. Ntarmos, P. Triantafillou, and G. Weikum. 2009. Distributed hash sketches: Scalable, efficient, and accurate cardinality estimation for distributed multisets. ACM Trans. Comput. Syst. 27, 1, Article 2 (February 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.

[PES+10] 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.

[RFC2330] V. Paxson, G. Almes, J. Mahdavi, M. Mathis, Framework for IP Performance Metrics, http://www.rfc-editor.org/rfc/rfc2330.txt , May 1998

[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.

[Sin84] R. W. Sinnott. Virtues of the Haversine. Sky and Telescope, 68:159, 1984.

[SM06] Phillipa Sessini and Anirban Mahanti, Observations on Round-Trip Times of TCP Connections, In Proc. of SCS SPECTS 2006. Calgary, Canada, July 2006

[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.

Copyright © ENVISION Consortium, February 2012

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

[TFK+11] 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.

[THT+12] Stefano Traverso, Kévin Huguenin, Ionut Trestian, Vijay Erramilli, Nikolaos Laoutaris and Konstantina Papagiannaki, TailGate: Handling long-tail content with a little help from friends, 21st International World Wide Web Conference, April 2012, France.

[TK06] Sergios Theodoridis & Konstantinos Koutroumbas (2006), Pattern Recognition 3rd ed. pp. 635.

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

[urla] Equinix. http://www.equinix.com/.

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

[WPD+10] 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.

Copyright © ENVISION Consortium, February 2012