Content aware searching, caching and streaming

Th. Zahariadis[1], F. Junqueira[2], L. Celetto[3], E. Quacchio[3], S. Niccolini[4], P. Plaza[5]

[1]Synelixis Solutions, Farmakidou 10, Chalkida, Greece
[2]Yahoo!Iberia, Av. Diagonal 177, Barcelona, Spain
[3]STMicroelectronics, C. Olivetti 2, Agrate, Milan, Italy
[4] NEC Europe, Kurfuersten-Anlage 36, Heidelberg, Germany
[5]Telefonica I+D, E. Vargas 6, Madrid, Spain

Abstract

The Future Media Internet is expected to be dominated by content-oriented traffic, including but not limited to P2P traffic, which does not conform to the client-server paradigm and generates unnecessary indirection overheads when users try to retrieve the desired data. In this paper, we describe the approach we are following in the EU project COAST in order to face the increasing need for a content centric internet. We aim to build a Future Content-Centric Network (FCN) overlay architecture able to find the desired data in the closest networking cache and forward it to the users in an efficient, timely and network-friendly way.

Keywords

Distributed Searching, Content Centric Internet, Netwok Architecture

1. Introduction

While the current Internet architecture is designed based on the client-server communication paradigm, the Future Media Internet is expected to be dominated by content-oriented traffic. Recent reports show that most of the Internet traffic is dominated by P2P traffic, which does not conform to the client-server paradigm and generates unnecessary indirection overheads when users try to retrieve the desired data. The inconsistency between the Internet design and the real usage is expected to be further increased, as the Future Internet is envisaged to provide the means to share and distribute (new) multimedia business and user-centric services, with superior quality and striking flexibility from everyone to everyone. In consequence, it is necessary to redesign the Future Internet based on a content-centric paradigm to provide data/content to the users in an efficient manner.

In this paper, we describe the approach we are following in the EU project COAST “COntent Aware Searching, retrieval and sTreaming” in order to face the increasing need for a content centric internet. We aim to build a Future Content-Centric Network (FCN) overlay architecture able to intelligently and efficiently link billions of content sources to billions of content consumers, and offer fast content-aware retrieval, delivery and streaming, while meeting network-wide Service Level Agreements (SLAs) in content and services consumption. In short, we foresee an FCN overlay network, where the users will just specify which content or service they need, and the framework will have the content and network awareness in order to find the desired data in the closest networked cache and forward it to the users following the “best” available and network-friendly way.

2. Current Internet content delivery limitations

In order to better explain the approach, initially we review how content discovery, retrieval and delivery take place in today’s Internet and where the proposed framework innovates (*Zahariadis et.al. 2010*). Today, the vast majority of Internet usage is data retrieval, data delivery/streaming and Web services access, where the user cares about content and is oblivious to their location. That is, the user knows that he/she wants news from CNN, videos from YouTube or weather information, but does not know or care on which machine the desired data or service resides. The above functionality is realised by the network architecture as shown in Figure 1. The network consists of: a) *Content Servers* or *Content Caches* (either professional or user generated content and services), b) centralised or clustered *Search Engines*, c) core and edge *Routers* and optionally *Residential Gateways* (represented as R1 to R5) and d) Users connected via fixed, wireless or mobile terminals.

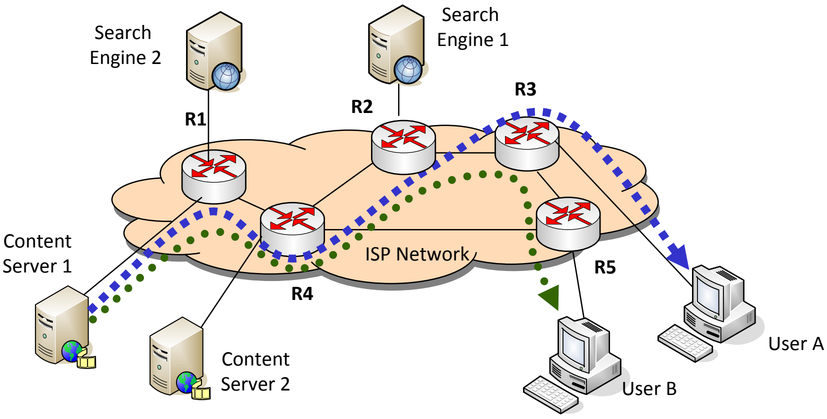

Figure 1 - Today’s Internet Architecture

The initial step is *Content Discovery by the Search Engines*: the Search Engines crawl the Internet to find, classify and index content or services. Alternatively, users may publish content and manually inform the search engine. The second step is *Content Discovery by the User*: the user queries a Search Engine and gets as feedback a number of URLs, where the content is stored. The last step is *Content Delivery/Streaming*: the user selects a URL and the content is delivered or streamed to him.

Yet in order to prove the inconsistency of today’s Internet let consider the simple case of delivery of a famous video from Content Server 1 (e.g. a YouTube server). The same video will be streamed a few dozens of times per square blocks. If a neighbourhood has a few dozens of square blocks, and a city a few hundreds of neighbourhoods, the same video already reaches some thousands of streaming. If we continue aggregating at country and world-wide level, we soon run out of existing bandwidth just for one popular video stream.

In the above scenario, if both User A (UA) and User B (UB) ask for the same content to the same Search Engine, they will both get as an answer that the content is stored at Content Server 1 (CS1). Then the content will be delivered via the routers’ (Rx) path: for example

CS1-R1-R4-R2-R3-UA and CS1-R1-R4-R2-R3-R5-UB respectively. Yet, all three steps of content discovery and delivery can be significantly improved:

- If the *content could be stored/cached closer to the end users*, not only at the end-points as local proxies, but transparently in the network (routers, servers, nodes, data centres) then content delivery would have been much more efficient.
- If the *routers could identify/analyse what content is flowing through them* and are able to *replicate* it efficiently, the search engines would gain much better knowledge of (even the streaming) content location and provide information even on "live" video streams. For example, if R2 could report to Search Engine 1 that "channel 5" content flows through it, when User B asks for "channel 5" Search Engine 1 could send R1 instead of CS1.
- If the *network could dynamically identify what is the best end-to-end path* (less congestion, lower delay, more bandwidth), it would have provided a better way to deliver the data. For example the path R2-R4-R5-UB may be much better than R2-R3-R5-UB.
- If the *content could be interactively adapted*, not only statically based on the network and terminal capabilities, but also based on the interactive content selection by the user (e.g. instantly changing the point of view, zoom-in/zoom-out at a streaming session), the user experience would be much better.

3. Proposed Content-Aware Networks Architecture

In the proposed architecture, we move intelligence in the network and adapt the Network Architecture of Figure 1, into a pure content-centric architecture as shown in Figure 2. The proposed network architecture consists of different virtual hierarchies of nodes (overlays), with different functionality. In the figure only 3 layers are shown; yet, this model may be scaled to multiple levels of hierarchy (even mesh instantiations, where nodes may belong in more than one layers) and multiple variations, based on the available level of information and service delivery requirements and constrains.

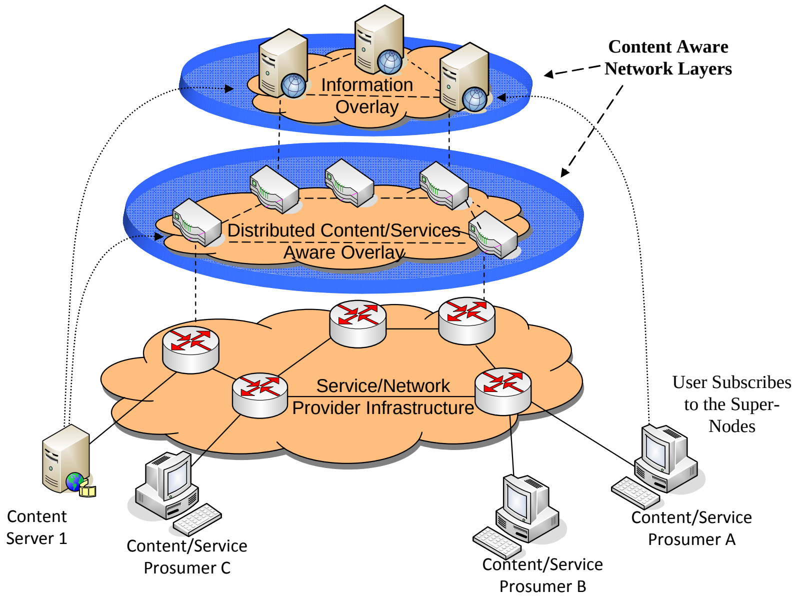

Figure 2: Proposed Content-Aware Network Architecture

At the lower layer, it is the ***Network Provider Infrastructure Overlay***. Users are considered as Content Producers and Consumers (Prosumers). In a realistic roll-out scenario, the FCN deployment will be incremental. Thus, today's existing legacy network nodes (core routers, switches, access points) will be the majority for a number of years. This Network Infrastructure Overlay is the service, ISP and network provider network infrastructure, which will consist of nodes with limited functionality and intelligence. Content is routed, assuming basic quality requirements and if possible cached to some degree in this layer.

The medium layer is the ***Distributed Content/Services Aware Overlay.*** Content-Aware Network Nodes (e.g. edge routers, home gateways, terminal devices) are located at this overlay. These nodes have the intelligence to filter the content and Web services that flow through them (via Deep Packet Inspection, DPI), identify streaming sessions and traffic (via signalling analysis) and provide qualification of the content. This information is reported to the higher layer of hierarchy. Additional virtual overlays (not shown in the figure) may be considered or dynamically constructed at this layer. We may consider overlays for content caching, content classification (even content indexing in the future), network monitoring, content adaptation, optimal delivery/streaming. With respect to content delivery, nodes at this layer may operate as hybrid client-server and/or peer-to-peer (P2P) networks, according to the delivery requirements. As the nodes will have information about the content and the content type/context that they deliver, beyond state of the art hybrid topologies may be constructed, customised for streaming complex media such as SVC, MDC and MVC.

At the highest layer, it is the Content/Services ***Information Overlay***. It will consist of intelligent nodes or servers that have a distributed knowledge of both the content/web-service location/caching and the (mobile) network instantiation/ conditions. Based on the actual network deployment and instantiation, the service scenario, the service requirements and the service quality agreements, these nodes may vary from unreliable peers in a P2P topology to secure corporate routers or even Data Centres in a distributed carrier-grade cloud network. The content may be stored/cached at the Information Overlay or at lower hierarchy layers, though the *Information Overlay* will be always aware of the content/services location/caching and the network information. Based on this information, it may decide on the way that content will be optimally retrieved and delivered to the subscribers or inquiring users or services.

3.1. Distributed Content & Services Searching

Within the proposed *Information Overlay*, we adopt the approach of a combination of distributed servers, sites and data centres, which enable search over the Internet-scale content collection, for a user base that comprises billions of users worldwide. In fact, we need a scalable solution that enables the addition of any new node, site or data centre at any time, independent of the size. In the following we use the term "site" to define one of the alternatives. We therefore intend to work on solutions for crawling, indexing, and query processing that fully utilize the capacity of multiple distributed sites efficiently. By efficiently, we mean minimizing the amount of resource consumption per query, maximizing the overall performance of the system, and yet providing a good user experience.

FCN will perform crawling in two ways: *actively* and *passively* (Figure 3). Active crawling (Figure 3a) consists of the traditional method of crawling, where crawlers (servers performing crawling) fetch documents from the Internet, index their information, and follow links in this content. Although there are solutions to distributed crawling, they are unsatisfactory for a large-scale search engine, mainly because it is difficult to avoid duplication and communication between sites. To split the crawling across different sites so that we serve users best, we have to analyze traffic logs to determine optimal or near-optimal load distribution.

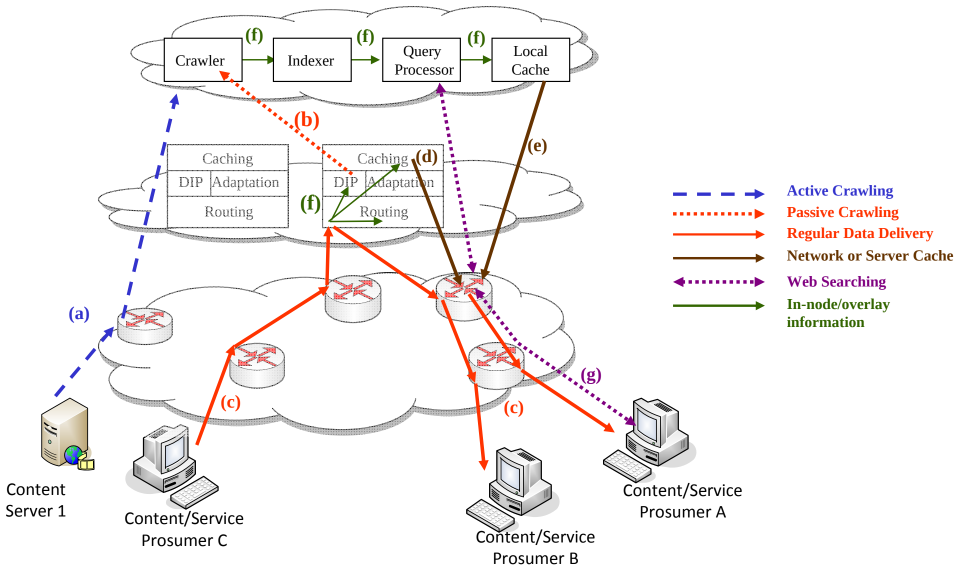

Figure 3 - Summary of search features and tasks.

With passive crawling, the search engine receives indexable information out of *deep-packet inspection* (DIP) performed by COAST intelligent nodes in the content/services overlay. Without interrupting the regular data delivery (Figure 3b), when information flows via an intelligent node, it crawls the data, cache them (if possible) (Figure 3c) and report to the Information Overlay interesting content and new Web services in the form of metadata (Figure 3d). As examples of metadata and different mechanisms, we have tags and URLs. With ***tags***, intelligent nodes just tell to the search engine what it has to index. With *URLs*, the search engine may inspect the content itself and decide if and how it will index the content. We intend to explore both possibilities and other innovative ones. As for Web services, we can detect, for example, *WSDL* descriptors in network packets and report those to the search engine. However, it is important to verify, whether such a discovery mechanism is effective and if there are better approaches that we can use along with DPI.

To evaluate crawling, there are several metrics such as throughput (number of documents per second). However, in the proposed platform, we are more interested in the quality of the content indexed. That is, we are interested in assessing how efficient our techniques are with respect to presenting the user with more relevant content. We consequently plan to evaluate the efficiency of our crawling process not only with systems metrics, such as throughput, but also with quality metrics that assess the quality of the content like *coverage*, *diversity*, and *freshness*.

3.2. Caching

Content caching may drastically increase network efficiency and lower the delivery delay. Moreover, caching of queries results and parts of the index will improve the efficiency of the FCN by avoiding performing the same computation twice. We can cache content, query results and parts of the index, not only at the Information Overlay, but subsets of them also at the intelligent nodes of the *Distributed Content/Services Aware Overlay*. Thus, we plan to explore policies for both types of cache.

Intelligent nodes may also cache content and the engine can point users to the closest cached copy (Figure 3g). Consequently, the engine consequently maintains information about cached copies and returns results pointing to cached copies accordingly. To have such information, sites either receive from intelligent nodes periodic snapshots of their cached content, or they decide what intelligent nodes have to cache by pushing content into the intelligent nodes. Which approach is the best depends on practical factors including the ability of the engine to decide effectively where to cache content and the inherent cost of collecting and maintaining information to make such decisions.

3.3. Content-aware Delivery

The proposed FCN is expected to allow smooth delivery of media content. This process will be based on a content-aware paradigm that exploits the information available in the FCN, as well as the FCN architecture, in order to organize, optimize and deliver the content. It will be managed by the Content/Services Aware Overlay, exploiting information available through the Information Overlay and the Service Provider Network. The key factors enabling content-aware delivery are:

Distributed media identification. The FCN will perform distributed media storage through its content indexing/caching services. The FCN will leverage on this distributed architecture, exploiting the knowledge available at the Information Overlay to optimally route the content to the users. Proper interfacing with the Information Overlay, as well as information coming from DPI will provide the parameters necessary to define the distribution policy.

Content-aware FCN probing. Content-awareness will allow not only organizing, but also optimizing and performing the actual content delivery process. To this end, the delivery policy will exploit distributed knowledge of the status of the content-aware network nodes, and of the extent that they can contribute to the delivery in either client-server or overlay mode. The first step will be the definition of a set of metrics that characterize a content-aware node, e.g. upload contribution in the last period of time, experienced download rate, level of the playout buffer, number of lost/delayed media segments. The second step will be the definition of a simple model able to estimate the Perceived Quality of Service (PQoS) offered to the end user and to assess the performance of the overlay at a global level, thereby estimating the available delivery policies for new sessions. Note that today it is very difficult to acquire information about where nodes are located in the network and the costs to reach them, because of complexity, network operational costs, or network policies that are not disclosed in detail. In the proposed FCN, this information will be estimated using active measurements between the content-aware nodes and information coming from DPI, and used to compute the delivery policy, ultimately improving performance and PQoS while reducing resource consumption in the underlying network infrastructure.

Matching content and network. Network- and content-awareness will be matched so as to compute a delivery policy, and to subsequently prepare the content in an adaptive way. The delivery policy will be based on the type of content, its current availability and location in the network, delivery requirements and SLAs, as well as on the nodes that require the content and their characteristics. Moreover, it will exploit the features of rich media representations such as SVC/MDC/MVC, augmented by on-the-fly content enrichment, so as to achieve maximum flexibility and adaptation.

Content-aware streaming. The actual streaming process will be based on bandwidth-efficient and low-delay strategies that allow to satisfy the PQoS requirements for the media and to meet the proper SLA. In a P2P environment, every node needs to know which segments are owned by its peers and explicitly pulls the segments it needs. Such systems offer good resilience to node failures, but involve overhead, due to the exchange of buffer maps between nodes (content reconciliation) and to the segment pulling process. We propose to explore different coding strategies for network coding, significantly improving streaming performance and reducing start-up times.

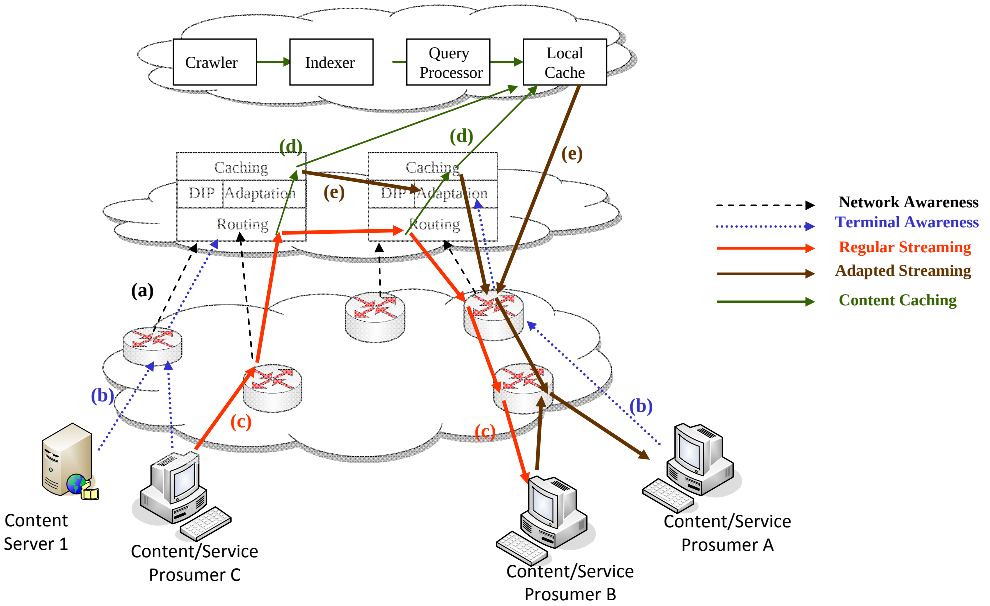

Figure 4 - Network/Terminal Awareness, Content Streaming & Adaptation

As it is shown in Figure 4, on regular intervals or on-demand, the intelligent nodes gain network and traffic awareness (Figure 4a) and terminal awareness (Figure 4b). During normal streaming (Figure 4c) or active crawling, as already explained, the distributed content/service aware overlay, gains knowledge of the content and Web services, and caches content locally (Figure 4d). When a new content user (Prosumer A) requests the same content or decides to join an active stream, cached content is streamed either from local or distributed caches (Figure 4e) and/or other Prosumers (e.g. Prosumer B), adapted to the network conditions, terminal characteristics or personal preferences.

4. Conclusions and next steps

In this paper, we describe the approach we are following in the EU project COAST in order to face the increasing need for a content centric internet. We aim to build a Future Content-Centric Network (FCN) overlay architecture able to find the desired data in the closest networking cache and forward it to the users in an efficient, timely and network-friendly way.

The above architecture will be tested in worldwide testbed based on Planetlab/OneLab and on TID's testbed. As it is shown in Figure 5 - COAST a) large scale testbed, b) Spanish/TID testbed

, test sites will be built in TID, Yahoo, NEC, TUB, HHI, Polito and UCLA. The different test sites will be fully interconnected with experimental facilities to test also the advances in the COAST Future Content Network.

Moreover, four TID sites will be interconnected to construct the COAST final user services testbed. Each site will contribute with a Digital Home demonstrator that has been built over the last few years. The objective is to demonstrate in almost real home environments the scenarios that will be developed during the lifetime of the project based on the fact that residential users will be sharing and streaming different media content over the internet. Moreover, the TID user experience lab will be used to evaluate real users, who interact freely with services.

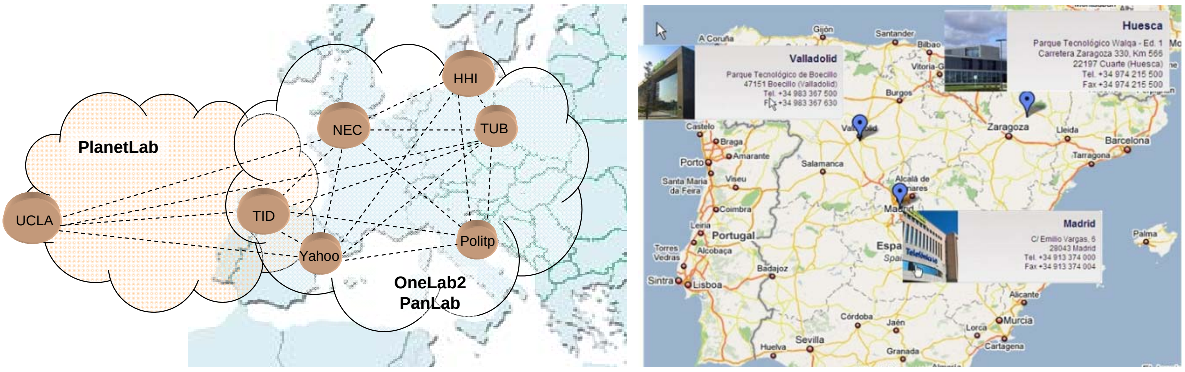

Figure 5 - COAST a) large scale testbed, b) Spanish/TID testbed

5. Acknowledgement

This publication is based on work performed in the framework of the Project COAST-ICT-248036, which is partially funded by the European Community. The authors would like to acknowledge the contributions to the COAST Project of colleagues from STM, Synelixis, Yahoo, NEC, TID, Fraunhofer HHI, Technische Universitaet Berlin, Politechnico di Torino, Fundacio Barcelona Media Universitat Pompeu Fabra, University of California, Los Angeles, Seoul National University.

6. References

Zahariadis Th., Daras P, Bouwen J., Niebert N., Griffin D., Alvarez F., Camarillo G., (2010) “*Towards a Content-Centric Internet*,” Towards the Future Internet, IOS Press, 2010, pp.227-237

Future Media and 3D Internet Task Force (2009), “*Future Internet and NGN: Design requirements and principles for a Future Media and 3D Internet*,” Networked Media Unit, February 2009

Clark D., Wroclawski J., Sollins K., and Braden R. (2002) “*Tussle in Cyberspace: Defining Tomorrow's Internet*”, 2002 Conference on Applications, Technologies, Architectures, and Protocols For Computer Communications, SIGCOMM '02, (2002), New York, NY, 347-356

Future Content Networks Group (2009), “*Why do we need a Content-Centric Internet? Proposals towards Content-Centric Internet Architectures*,” White paper. 2009