COntent Mediator architecture for content-aware nETworks

European Seventh Framework Project FP7-2010-ICT-248784-STREP

Deliverable D2.2
High-Level Architecture of the COMET System

The COMET Consortium

Telefónica Investigación y Desarrollo, TID, Spain
University College London, UCL, United Kingdom
University of Surrey, UniS, United Kingdom
PrimeTel PLC, PRIMETEL, Cyprus
Warsaw University of Technology, WUT, Poland
Intracom SA Telecom Solutions, INTRACOM TELECOM, Greece

For more information on this document or the COMET project, please contact:

Prof. George Pavlou
g.pavlou@ee.ucl.ac.uk
Department of Electronic & Electrical Engineering
University College London
Torrington Place, London,
WC1E 7JE
UK

Document Control

Title: High-Level Architecture of the COMET System

Type: Public

Editor(s): Ioannis Psaras

E-mail: i.psaras@ee.ucl.ac.uk

Author(s): Ioannis Psaras, George Pavlou (UCL)

Gerardo García de Blas, Francisco Javier Ramón Salguero (TID)

Ning Wang (UniS)

Eleftheria Hadjioannou (PRIMETEL)

Andrzej Beben, Jaroslaw Sliwinski, Jordi Mongay Batalla, Wojciech Burakowski (WUT)

Sergios Soursos, Vasiliki Kamariari and George Petropoulos (INTRACOM TELECOM)

Doc ID: D2.2-v1.0.doc

AMENDMENT HISTORY

Version	Date	Author	Description/Comments
V0.1	September 27th, 2010	Ioannis Psaras	First version, ToC
V0.2	October 5th, 2010	Ioannis Psaras	Second Version, ToC
V0.3	October 6th, 2010	IoannisPsaras	Final Version, ToC
V0.4	October 29th, 2010	Ioannis Psaras	First draft of COMET Architecture
V0.5	November 6th, 2010	Ioannis Psaras	First review round of the COMET Architecture
V0.6	November 13th, 2010	Ioannis Psaras, Wei Koong Chai	Integration of the Coupled Approach description
V0.7	November 17th, 2010	Ioannis Psaras	Integration of partners' comments
V0.8	November 23rd, 2010	Ioannis Psaras, Marinos Charalambides	Input to Section 2.2 and 3.3.1
V0.9	December 20th, 2010	Ioannis Psaras, Gerardo García de Blas	Draft Final version to be submitted to the internal reviewers
V0.10	December 23rd, 2010	Ning Wang, Francisco Javier Ramón Salguero	Modifications and comments from the internal reviewers
V0.11	January 3rd, 2011	Gerardo García de Blas, Ning Wang	Final version to be reviewed by the Consortium and to be submitted to the EC
V1.0	January 10th, 2011	Gerardo García de Blas	Final version of the deliverable

Table of Contents

1 Executive Summary

The purpose of this document is to define the *High-Level Architecture of the COMET System*. This architecture includes all the properties required to accommodate the appropriate content naming, resolution and delivery mechanisms for the COMET system. The details of these mechanisms are included in the subsequent deliverables D3.1: *Interim Specification of Mechanisms, Protocols and Algorithms for the COMET Mediation System* [2] and D4.1: *Interim Specification of Mechanisms, Protocols and Algorithms for Enhanced Network Platforms* [3].

The COMET approach will make access to content both location- and application-independent, supporting discovery, access and content/network-aware distribution for all types of content. Central to the COMET concept is the definition of a global naming scheme and the optimization of both content source selection and content distribution, by mapping the content to the appropriate network resources based on content transmission requirements, user preferences and network state.

To achieve the previous goal, COMET follows a two-plane approach. The upper plane is the ***Content Mediation Plane*** (CMP), which is mainly responsible for *name* and *content resolution as well as the preparation of the path used for content delivery*. That is, upon a user's request, the CMP has to find the best available copy, based on various criteria, including content QoS class requirements and more dynamic information such as server load, path conditions etc.

The lower plane of the COMET system is the ***Content Forwarding Plane*** (CFP) and is mainly responsible for delivering the content back to the content consumer. This is done based on instructions provided by the CMP, based on the information about server and path conditions. Therefore, it is clear that the two planes work in parallel and are not completely decoupled from each other, operation-wise. That is, the upper plane, i.e., the CMP, has to provide information to the lower plane, in order to enable network-awareness and smooth delivery of content to the Content Consumer.

The aforementioned planes collaborate with each other in order to accomplish the two main COMET operations, namely the *Content Publication* and the *Content Consumption*, as described in D2.1 *Business Models and System Requirements* [1].

In this document, it is initially introduced the concept of *Content Mediation as Key Enabler for Content Distribution*. The mediation lies in the provision by Internet Service Providers (ISP) of an intermediate plane between the world of contents and the world of data transmission. On the one hand, this mediation plane allows the ISPs to be aware of the content characteristics, the content sources and their performance requirements, while being aware of the network and its conditions. On the other hand, the mediation plane allows ISPs to act as mediators for content distribution, offering a common interface for content consumption, while instructing the network in order to improve content delivery in terms of quality and effective bandwidth utilization. More details about the concept of content mediation are given in Chapter 2. The chapter begins with a discussion of the problems with current content distribution approaches, given as the *Problem Statement*, then the *Rationale of the Content Mediation Approach* is presented, and, finally, the *Benefits of the Network and Content Mediation* are discussed.

Next, in Chapter 3, the COMET architecture is described, basing its design on an *Analysis and Synthesis Methodology*. We start from the basic COMET operations to be realized and elaborate on the *processes* done by the aforementioned planes. These processes are grouped into *functional blocks*, which are the main building blocks of the *Content Mediation Plane* and the *Content Forwarding Plane*. They include functions such as *Content Mediation, Content Resolution, Path Management, Content-Aware Forwarding*, and *Server and Network Monitoring*.

Based on the overall COMET architecture, two approaches are described and studied, namely the *Decoupled* and the *Coupled Content Mediation Approaches*. The first approach (Decoupled) follows the basic paradigm of the current Internet by allowing the physical signalling routes for content resolution and content delivery to be separated but coordinated. This allows a more graceful transition from the current host-centric Internet to the content-centric one as it generally abides to the DNS system we use today. The second approach (Coupled) takes on a more involved path whereby both the content resolution and content delivery are coupled.

Further on in Chapter 3, it is also clearly drawn the line between WP3 and WP4 of the COMET Project and therefore, of the details to be included in D3.1 [2] and D4.1 [3], respectively.

In Chapter 4, the four use cases defined in D2.1 [1] are mapped to the COMET Architecture. Each use case is described in terms of specific steps and interactions between the COMET entities and the functional blocks.

Finally in Chapter 5, details are given on how the COMET Architecture covers the System Requirements that were defined in D2.1 [1]. There it is shown that, as required, more than 70% of these requirements are covered by the architecture described in this document. Besides, details are given for those requirements that are not yet covered by the current architecture, building our future work plan for the 2nd and 3rd years of the project.

2 Introduction: Content Mediation as Key Enabler for Content Distribution

The *host-centric* nature of the current Internet causes two main problems:

1. **While users are requesting for content or services, the Internet directs these requests to the machines that host the corresponding content or are responsible for accomplishing the service**. As a result, networks are unaware of the content they are transporting and network operators cannot apply the most appropriate end-to-end transport strategy for each content.
2. **Global content search and resolution is fragmented** due to the lack of a unified content naming architecture.

The above two problems have several implications that hamper the operation of today's networks. For example, the first problem depicts a clear conceptual mismatch between theory and practice. The second problem has resulted in a huge number of intermediaries and communities the user needs to be a member of in order to access content. Clearly, this fragments both the search and the resolution of content: although multiple copies of the same content may exist in the Internet they can be accessed by members of the corresponding communities only. This results in both inefficient content search and limited availability of content from all interested content clients.

Despite the aforementioned conceptual mismatches under which content access, resolution and delivery are taking place today, the Internet can still cope with users' demand; there is, however, plenty of room for improvement of the users' quality of experience and optimization of server and network resource usage. This has resulted in the need for massive over-provisioning from the ISPs' point of view, for instance. However, the unprecedented growth of user-generated content is expected to expand even more through both popular social networking sites (e.g., Facebook, MySpace etc.) and through individual platforms (e.g., blogs, personal streaming servers). This will push things further to a point where over-provisioning is neither efficient, nor cost-effective any more. The conceptual mismatch will therefore affect all the involved players of the content-based network, as identified and discussed in *D2.1: Business Models and System Requirements* [1]:

- *Content Creator*: the entity (individual or organization) that owns the rights to the content and wants to publish it on the Internet;
- *Content Provider* (which can also be the *Content Creator)*: the entity responsible for storing and making content available to the *Content Consumers* (usually a large organization, e.g., YouTube, Apple iTunes store);
- *Content Distributor* (which can also be the *Content Provider*): the entity that owns and maintains the infrastructure to distribute content to *Content Consumers* in the most effective way (e.g., CDNs, P2P networks);
- *Network Operator*: the entity that provides networking services, e.g., ISPs, or Internet Backbone Providers (IBPs);
- *Content Consumer*: the entity that consumes the content (usually the end-user). In case the end-user is both provider of content (e.g., user-generated content) and consumer of other contents, the term *content prosumer* is used.

The problems, as identified in D2.1 [1], for each of the players of the content-based network are summarized next. Content-agnostic transport, for example, makes it difficult for *Network Operators* to efficiently manage their networks, in terms of (web-) caching and replication as well as providing content-specific QoS. That is, due to the unpredictable traffic patterns that user-generated content brings, when demand exceeds supply, even over-provisioning cannot efficiently handle network and server resources. *Content Consumers* on the other hand do not have the option of accessing all available contents in the Internet due to the lack of a unified naming architecture, which further results to fragmented search and access, as mentioned earlier. Furthermore, users perceive low QoE (Quality of Experience), since the network usually offers only the best effort service without any delivery guarantees. Finally, *Content Creators* and *Providers* reduce the quality of the media they are producing (e.g., video) due to scarce network resources (e.g., bandwidth). For example, it is preferable to watch a video without disruptions but with low resolution, rather than to have a high resolution but disrupted playback.

The purpose of this document is to define the *High-Level Architecture of the COMET System*. This architecture includes all the required properties needed in order to accommodate the corresponding content naming, resolution and delivery mechanisms as realized in COMET. The details of these mechanisms are included in the subsequent deliverables D3.1: *Interim Specification of Mechanisms, Protocols and Algorithms for the COMET Mediation System* [2] and D4.1: *Interim Specification of Mechanisms, Protocols and Algorithms for Enhanced Network Platforms* [3].

The objective of the COMET system as a whole is to develop a ***unified content naming, addressing and resolution architecture***, where the user's request points to the content or service itself, rather than to the machine that hosts the content. In addition, ***server, network and routing awareness will inherently improve QoS for*** content consumers, based on the content requirements, rather than on holistic bandwidth over-provisioning, as happens today.

In the following, we present the Problem Statement (Section 2.1) and the corresponding rationale behind the COMET System design (Section 2.2) and finally, the benefits of the COMET mediation system (Section 2.3). Further discussions on these issues as well as the business models upon which our approach is based can be found in D2.1 [1]. Chapter 3 initially provides an overview of the methodology followed in order to construct the COMET Architecture. Then it elaborates on specific *operations, processes* and *entities* as these have evolved through the course of the project. We explain how the COMET Use-Cases defined in D2.1 [1] are mapped onto the COMET Architecture in Chapter 4 and finally, we describe how the architecture accommodates the COMET System Requirements, defined in D2.1, in Chapter 5 [1].

2.1 Problem Statement - Problems with current content-distribution approaches

In the current content distribution ecosystem, three main problems have been identified (see D2.1 [1]):

- In recent years there has been a growing proliferation of user-generated Internet content, including blogs, photos, video, etc. The increasing trend of users generating their own content has resulted in **global content search and direct access being fragmented**. Members of specific user-communities only can reach content on those communities, being difficult and sometimes impossible for other interested users to reach the desired content. There is common consent by now that there is no unified global content naming scheme to access the content and resolution architecture. This forces end-users to search the content through relevant intermediaries, maintaining a multiplicity of accounts, front-ends, tools and applications for content discovery and consumption.
- **Today's networks are unaware of the content they are transporting**, and, therefore, network operators cannot apply the most appropriate end-to-end transport strategy to provide the adequate quality of experience to the consumers. Moreover, even when the networks are well provisioned, intermediaries acting as Content Distributors (Internet Content Providers such as YouTube, CDN providers such as Akamai or P2P platforms such as Octoshape) cannot be aware of the network capabilities, traffic conditions, or the transmission requirements of the content. Therefore, the content is delivered far from the most efficient way.
- Due to the host-centric nature of the Internet, it has become apparent that while users are searching for content or services, the Internet, as a host-centric entity, points to the host or machine where the content is located rather than to the content itself. In addition, the

increased complexity of today's networks and the unprecedented growth of user-generated content results in either poor performance during content delivery, or inefficient utilisation of the existing network and server resources.

These problems stem from a common issue in the current Internet paradigm that should be highlighted as a relevant research challenge for the Future Internet: an appropriate link between the "worlds" of contents and data transmission is missing, thus hindering the effective coordination between them.

2.2 Rationale of the Content Mediation Approach

After analyzing in the previous section the problems that affect the content distribution ecosystem, we propose an approach for the Future Internet architecture based on the concept of mediation.

This mediation lies in the provision by Internet Service Providers (ISP) of an intermediate plane between the world of contents and the world of data transmission.

On the one hand, this mediation plane will act as mediator for content publication, offering an interface for content publication and thus becoming aware of content characteristics such as the content QoS requirements (content awareness), as well as the available content sources and their performance (server awareness). Moreover, since the mediation plane is provided by ISPs, it can be aware of the network topology and the available routing paths between content servers and clients (routing awareness), as well as the network conditions (network awareness). A rough outline of how this is achieved is shown in Fig. 1.

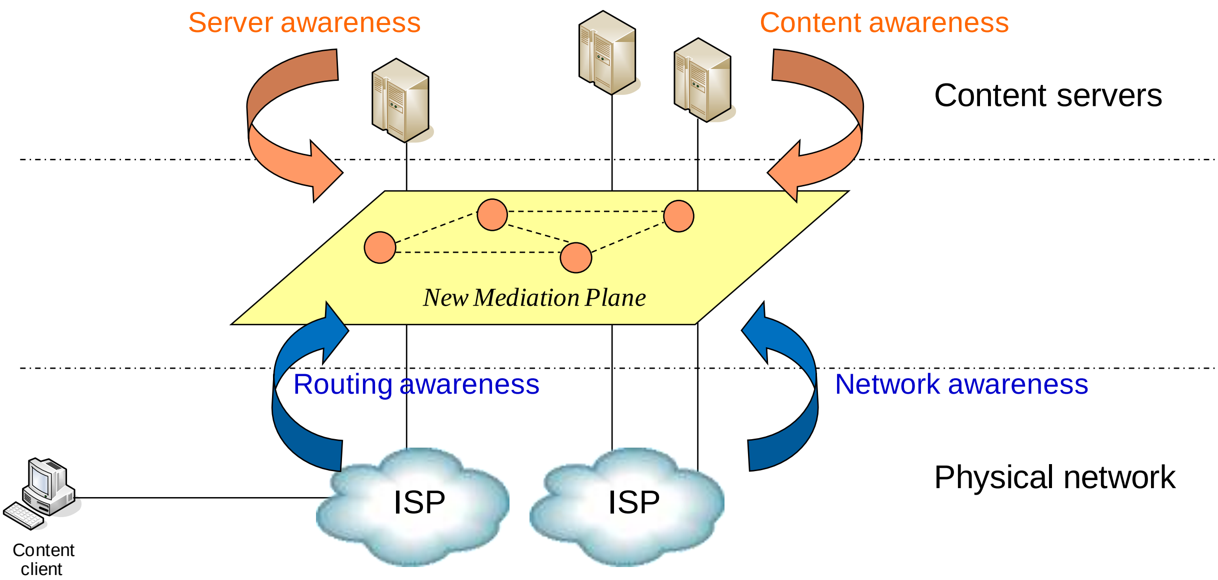

Figure 1. Awareness in the mediation plane

On the other hand, the mediation plane allows ISPs to act as mediators for content distribution, offering a common interface for content consumption, so that content is treated as a *first-class citizen* in the Internet. Thanks to the awareness achieved, ISPs will be able to decide the best sources and paths to deliver the content. Besides, since ISPs are naturally able to mediate with the network, it would be possible to instruct the network (network mediation) in order to improve content delivery in terms of quality and effective bandwidth utilization – see Fig. 2.

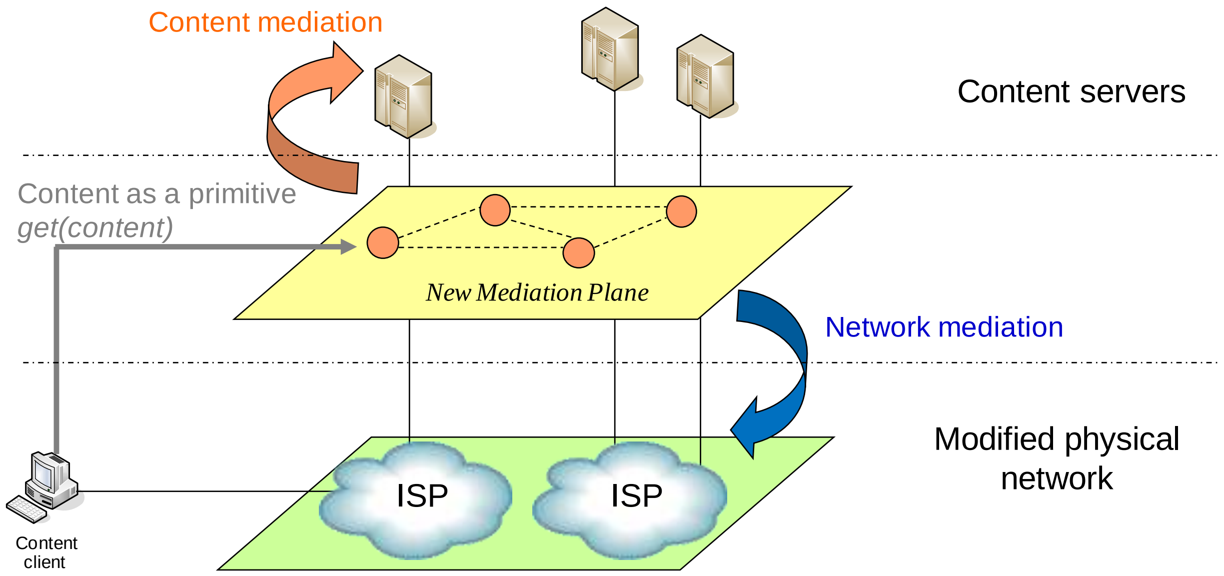

Figure 2. Mediation by the mediation plane

The key technical advantages that can be achieved thanks to this mediation are:

- Unified access to the content whatever its nature and location.
- Content delivery with guaranteed QoS.
- Point-to-multipoint content delivery capabilities, reducing bandwidth needs for live contents.
- Graceful handover of the content delivery path, providing more resilience and flexibility for multi-homed users.
- Advanced publication mechanisms, allowing Content Providers to update content servers on-the-fly, while switching among different ways of distribution.

2.3 Benefits of Content and Network Mediation

Mediation will benefit content resolution and delivery in a variety of ways, which differ among the various players of the COMET system. For example, *Network Operators* will reduce infrastructure costs by efficient resource allocation, e.g., optimal content server selection and deployment at the domain edges. Busy-link offloading through network awareness is another feature that network operators can take advantage of and further reduce infrastructure extensions, e.g., bandwidth over-provisioning. These techniques will also be complemented with caching mechanisms to increase and enhance the COMET system's resilience. These techniques are still under investigation and will be reported in later deliverables in the course of the project. Furthermore, efficient content delivery together with the appropriate business models will present *Network Operators* with the opportunity to provide new services that are not possible to provide today.

On the other hand, *Content Consumers* will benefit from the above, enhanced *Network Operator* functionalities with higher Quality of Experience. This may come in the form of improvement in data throughput, lower delivery delays, and increased security and reliability. In addition, as mentioned earlier, one of the main objectives of the COMET architecture/project is to produce a unified naming and resolution architecture, which will simplify the users' access to published content.

Finally, *Content Creators* and *Content Providers* will ultimately benefit from content mediation and awareness, since: i) content will be widely available, without search and access fragmentation limitations, ii) bandwidth requirements will decline due to both efficient caching/storage and different distribution schemes, such as point-to-multipoint, and iii) reliability and user reachability will increase due to more efficient services provided by the *Network Operators*. Moreover, the unified naming architecture will simplify the content publication and management to *Content Providers*.

The above issues are more extensively and comprehensively discussed in *D2.1 Business Models and System Requirements* [1].

3 High-Level Architecture of the COMET System

3.1 Architecture Design Methodology

The design of the COMET architecture follows the analysis and synthesis method.

The analysis starts with identification of the **operations** performed by the actors external to the COMET system (e.g., content publication and content consumption). Those operations usually require a number of **sub-operations** with clearly defined outcome (e.g., content storage, content registration, content resolution and content delivery). Each sub-operation may be self-contained or may be composed of a number of **processes**. We assume that processes are atomic and they cannot be split into smaller parts without going into implementation details. Since the focus of this deliverable is not the implementation details of the architecture, but rather its conceptual foundations, we omit discussions on such issues in this deliverable. In case of self-contained sub-operations, we assume for the sake of simplicity that the sub-operations have one process.

The next figure shows the analysis strategy, also known as top-down strategy, followed in the first step of design of the COMET architecture.

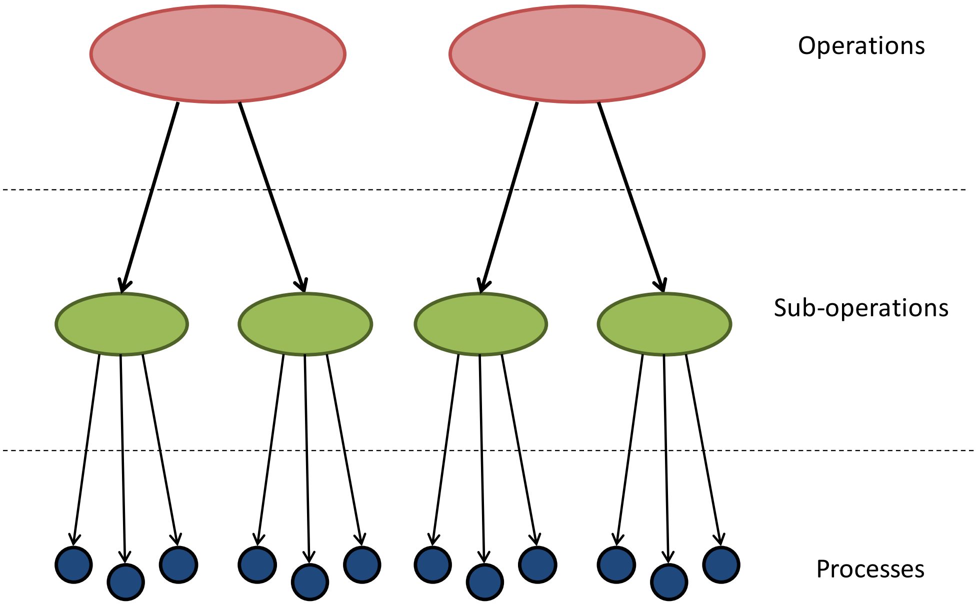

Figure 3: Top-down strategy: analysis of processes

The second stage in the design of the COMET architecture is the synthesis, which starts with the separation of all processes into two **planes**:

- Content Mediation Plane (CMP), which covers all processes related to handling the information about content (gathering content information, resolving, making decisions, etc.),
- Content Forwarding Plane (CFP), which provides abilities to deliver the content itself.

Next, we create **functional blocks** by gathering the processes that share and handle information of similar context. At this point, we complete the design of the COMET high-level architecture.

Figure 4 shows this synthesis strategy, also known as bottom-up strategy.

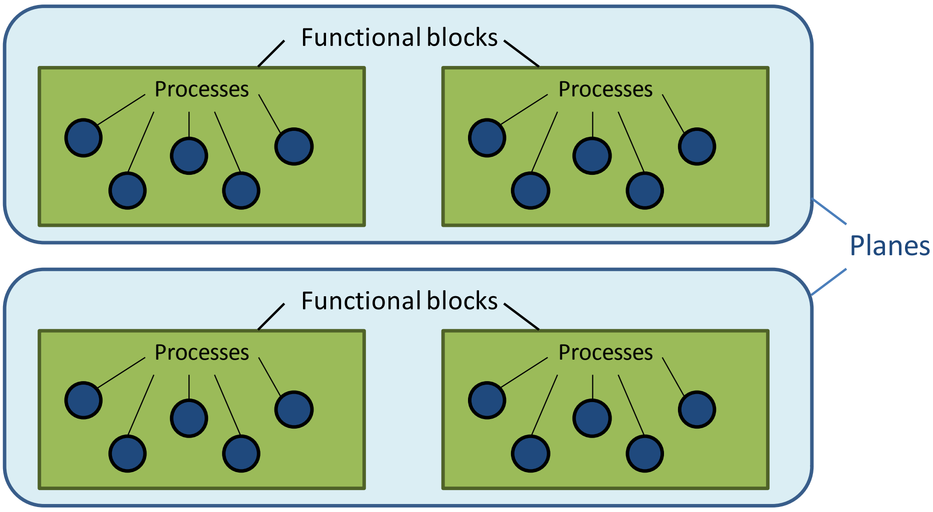

Figure 4: Bottom-up strategy: synthesis of processes into functional blocks and planes

3.2 COMET operations, sub-operations and processes

3.2.1 Top-down analysis from COMET operations to processes

We begin our discussion from the main operations that the COMET System must be able to offer. According to initial description in *D2.1 Business Models and System Requirements* [1], there are two main **operations** requested from the COMET System:

1. **Content Publication**: This operation is responsible for making a new piece of content available for access by Content Consumers. The initiator of this action is the Content Provider.
2. **Content Consumption**: This operation is initiated by the Content Client, who requests delivery of a particular content, using a specific content identifier.

Content Publication can be decomposed into 3 self-contained **sub-operations** (they do not split into multiple processes), which are depicted in Figure 5. In particular, they are:

i) **Content ID allocation**: The COMET System identifies the content by using Content-IDs or Content Names. Note that the allocation of the Content-IDs and Content Names may be arbitrary and it may not involve any interaction with COMET System. The details about selected allocation schemes are given in [2]. The outcome of this operation is a globally unique Content-ID and/or Content Name allocated for particular content.

ii) **Content storage**: The COMET System does not control the distribution of the content in Content Servers; this sub-operation is external to the COMET system. Nevertheless, the network location of the content copies must be known before the Content registration is performed. The outcome of this sub-operation is the placement of the content copy in particular network location and the correlation of the Content-ID with this specific location. This has to be made known to the COMET system, something that is done by the next sub-operation, namely the Content registration.

iii) **Content registration**: In this sub-operation, the COMET System creates a relation between Content-ID and network locations of the content copies. It can be performed only after the previous two sub-operations are completed. The outcome of this sub-operation is a situation where particular content may be used in Content Consumption.

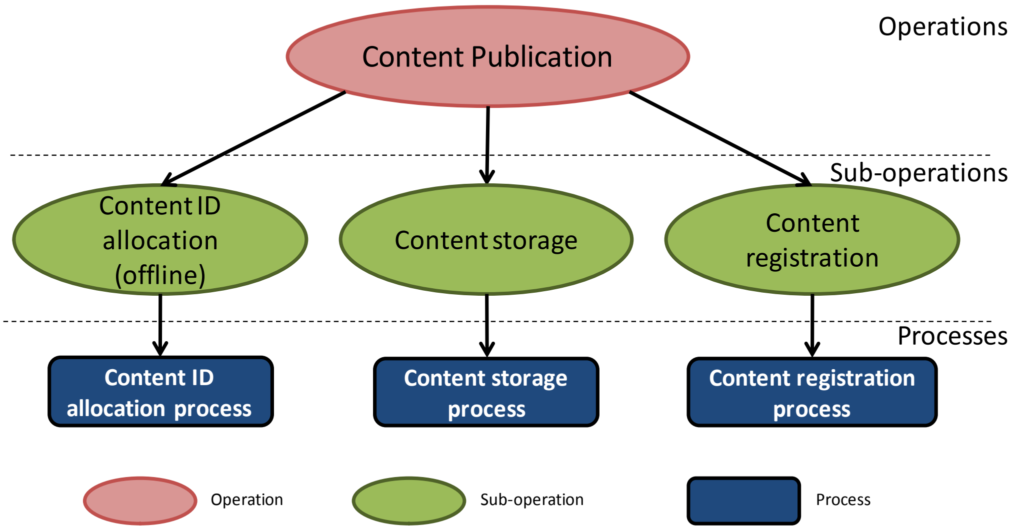

Figure 5: Top-down analysis of Content Publication operation

Content Consumption can be decomposed into 3 **sub-operations**, which are depicted in Figure 6:

i) **Awareness**: This sub-operation gathers, in an offline manner (it is not related to particular Content Consumption request), the current information about routing (topology), server conditions and network conditions. The outcome of this sub-operation is the knowledge, which is essential in making correct decisions during Content Consumption. Awareness is covered by 3 COMET **processes**:

   a. First, the process of **providing routing awareness** observes the network topology by new and existing mechanisms, e.g., business relations, BGP routing tables and other protocols for gathering of network reachability information.

   b. Second, the process of **providing server awareness** gathers the information about servers' conditions. This may include CPU load, number of active streams, traffic load over network interface cards, etc. Note that this process may interact with existing solutions for **server monitoring**. This server monitoring is considered external to the COMET system.

   c. Third, the process of **providing network awareness** gathers the information about conditions of links and/or paths in the network, e.g., link load and packet transfer delay metrics. This process may rely on and interact with specific solutions for network monitoring, external to the COMET system.

ii) **Content Resolution**: This sub-operation uses Content-ID and/or Content Name to prepare the network for Content Delivery sub-operation. The outcome of this sub-operation is the situation where the COMET System decides which server and which path prepared in the network should be used for particular Content Consumption. It is accomplished by four COMET **processes**:

   a. First, the process of **name resolution** locates the content by using Content-ID and/or Content Name.

   b. Second, the process of **path discovery** obtains, for each specific requested content, the properties of the paths from Content Servers to the Content Client, using the output of the process of providing routing awareness.

   c. Third, the **decision process** combines the results of above two processes to select the best server and path in the network for particular Content Consumption.

   d. Fourth, the process of **path configuration** is responsible for enforcing the decision at the network level.

iii) **Content Delivery**: The outcome of this sub-operation is a delivery of the content to the Content Client according to the decisions made during Content Resolution sub-operation.

This sub-operation contains only a **content forwarding** process, which involves low-level functions in the network nodes forwarding the content.

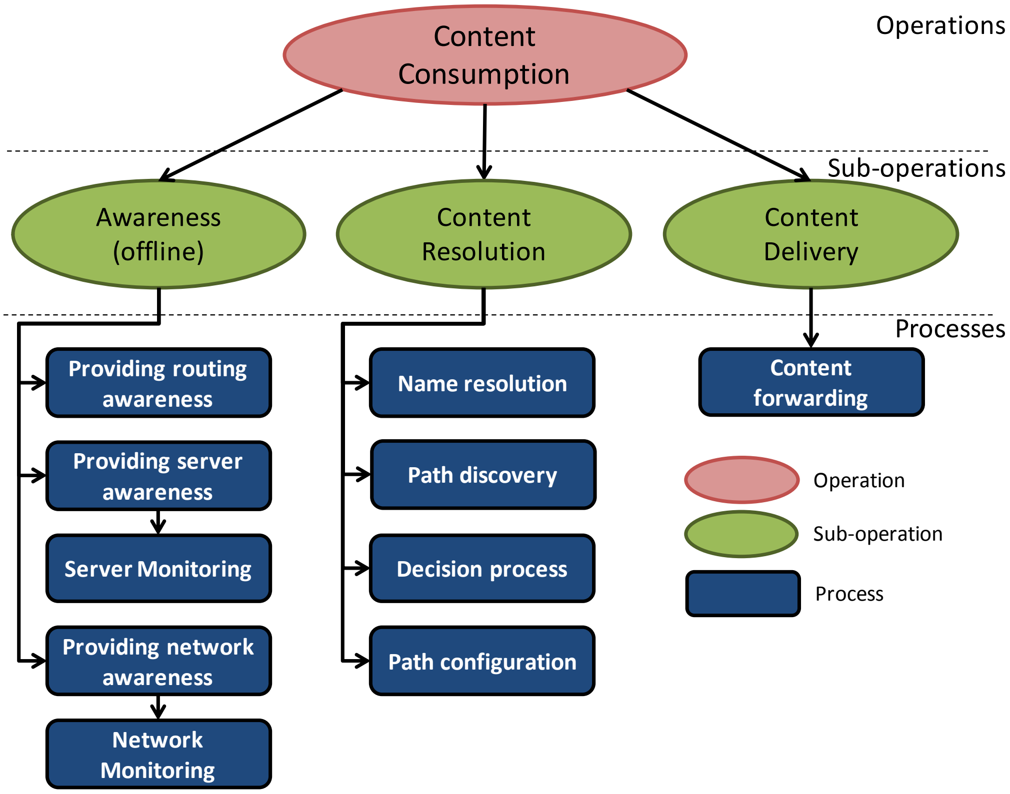

Figure 6: Top-down analysis of Content Consumption operation

3.2.2 Mapping of processes into planes and functional blocks

To achieve the above operations, COMET follows a two-plane approach. The upper plane is the **Content Mediation Plane** (CMP), which is mainly responsible for name and content resolution as well as the preparation of path used for content delivery. The lower plane of the COMET system is the **Content Forwarding Plane** (CFP) and is mainly responsible for delivering the content back to the content consumer. This is done based on mediation performed by the CMP taking into account the information about server and path conditions.

The COMET processes, which were identified in the analysis step, are grouped into five functional blocks. Figure 7 shows this grouping including the assignment to the CMP and CFP planes.

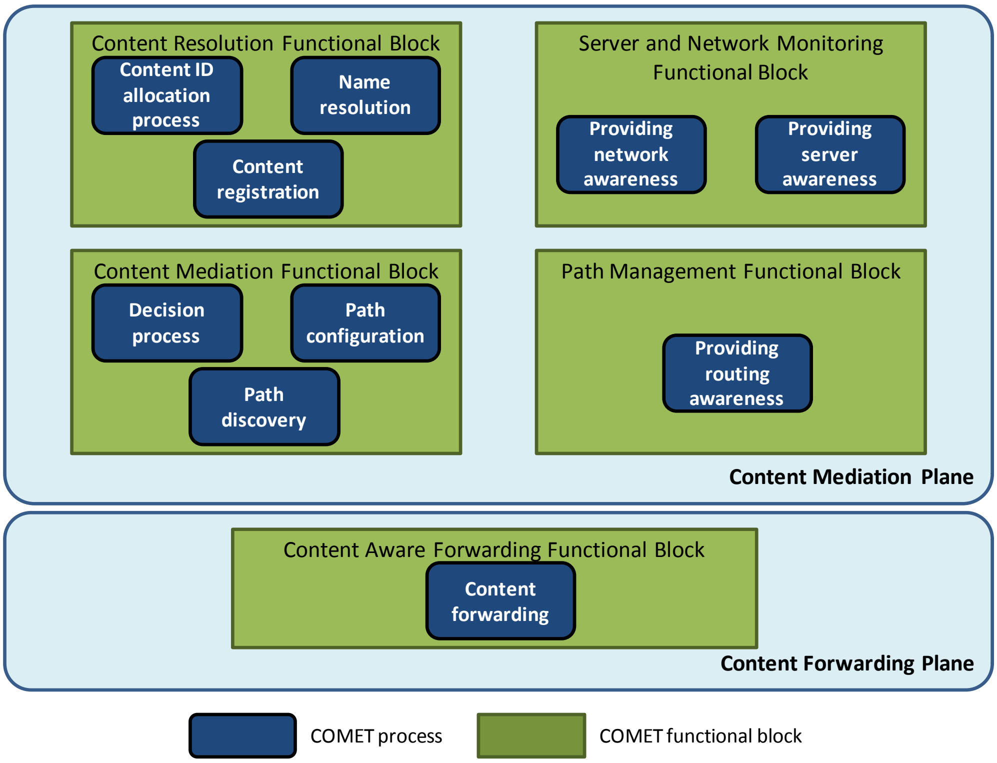

Figure 7: Bottom-up synthesis of processes into functional blocks

Next, Section 3.3 describes in detail these planes and functional blocks, as well as their relation with external entities (content client, content server and content publisher), which completes the COMET high-level architecture.

3.3 Overall COMET Architecture and the COMET Functional Blocks

Our investigations have shown that in order for our targets to be met, objective-wise, a large part of the COMET control functionalities have to be added to the upper plane, i.e., the *Content-Mediation Plane*, providing the *intelligence* needed for the network to become *content-aware*, while the CFP implements the data plane functionalities to deliver content according to instructions provided from the CMP. The details of the entities in the *Content Mediation Plane* are given in D3.1 [2], while the features added to the lower network level, i.e., the *Content Forwarding Plane*, CFP, is illustrated at D4.1 [3].

The COMET Architecture is composed of:

- ***Two planes***, the upper *Content Mediation Plane* (CMP) and the lower *Content Forwarding Plane* (CFP), as already discussed before,
- ***A number of functional blocks***, which accomplish the operations of the COMET system from content publication to content consumption. These functional blocks are grouped into one or more entities, as we explain later on in the present section.
- ***A number of entities*** (i.e., *Content Client, Content Server* and *Content Publisher*), which interact with the functional blocks in the COMET system in order to i) *publish content* (i.e., register content to the COMET system), ii) *request for content* and iii) *provide content-server monitoring information*, as depicted in Fig. 8.

In this section, we describe the basic *functional blocks* that build up the COMET architecture.

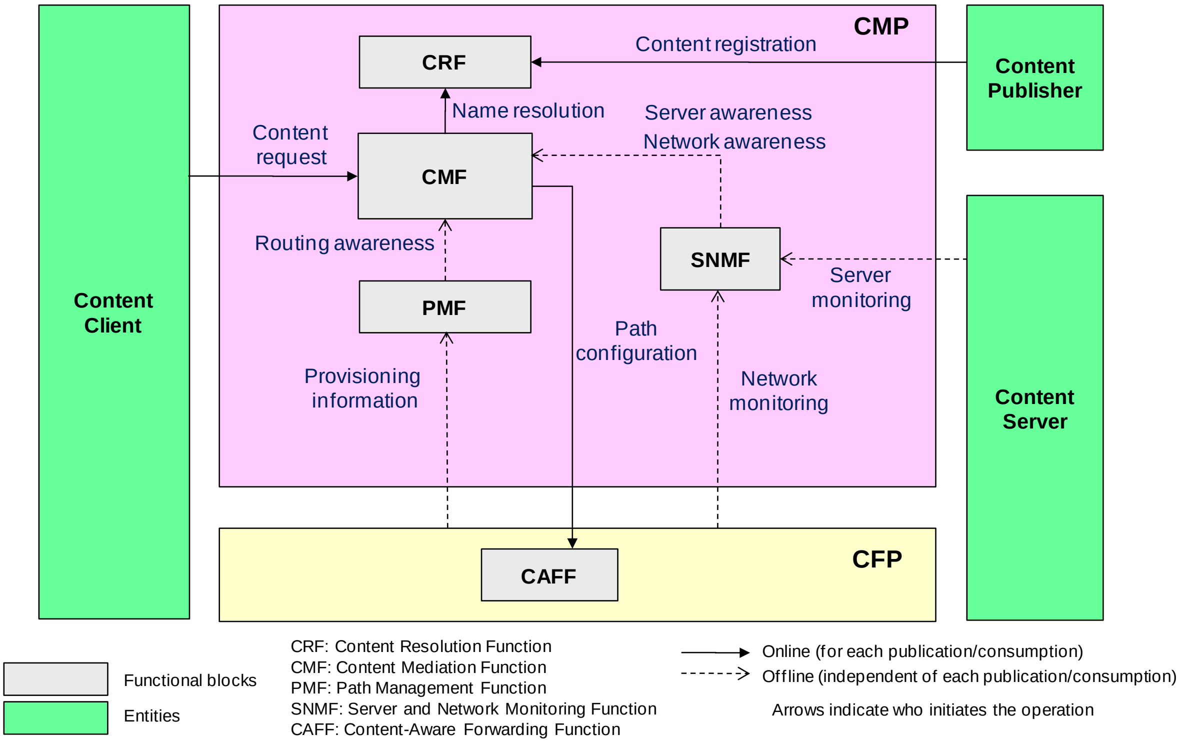

Figure 8. Overall COMET Architecture

The functional blocks included in the Content Mediation Plane are the following: the Content Resolution Functional block (CRF), the Content Mediation Functional block (CMF), the Path Management Functional block (PMF) and the Server and Network Monitoring Functional block (SNMF). On the other hand, the main functional block of the Content Forwarding Plane is the Content-Aware Forwarding Functional block (CAFF). Below, we discuss each of these functional blocks separately and we also give details on the interactions between them.

Content Mediation Plane

- **Content Resolution Functional block (CRF):** This function is triggered during the content publication process as well as content consumption operations. During the content publication operation, the CRF component is contacted by the *Content Publishers* every time a new content is to be published to the COMET system. Therefore, the main task of the CRF is to *keep track of the content records or indexes*. It is clear that for scalability purposes not all CRFs have knowledge of all the contents of the Internet/COMET system, but each CRF is keeping track of the records of its *area of responsibility*. This includes both allocation of content identifiers as well as the registration of new unpublished content. The second main responsibility of the Content Resolution Function is to *resolve Content Names to Content Properties*. These *Content Properties* may include *Content Records, metadata,* or even translation of the *Content Name* to *Content-ID*. In particular, as shown in Fig. 8, the content client triggers the *Content Mediation Functional block* (CMF), in order to request for content. Next, the CMF forwards the request to the CRF, in order for the latter to i) *resolve the Content Request* and ii) *locate the copies of the requested content*. The locations of the copies are returned back to the CMF. We discuss later on the responsibilities of the CMF when receiving this information. Further details on the *content resolution* sub-operation within COMET are given in D3.1 [2].

- **Path Management Functional block (PMF):** The PMF is one of the functional blocks of the CMP that interacts with the underlying network and the functional blocks belonging to the CFP. In particular, the PMF is responsible for gathering *Network Reachability Information*, which relates mainly to the underlying topology (i.e., inter-domain connections/links, link availability etc.); that is, routing information regarding the delay, loss and bandwidth (in terms of network load) of underlying network paths. This information is provided upwards to the Content Mediation Functional block (CMF).

- **Server and Network Monitoring Functional block (SNMF)[1]:** The responsibility of SNMF is pretty straightforward: it is responsible for collecting data for the status of: i) content servers, namely their *availability* and *load* and ii) the underlying network conditions, namely *ingress and egress load on peering links* (for multi-homing scenarios) and *load on access links* (e.g., for admission control). This information is then fed to the CMF for it to make the appropriate decisions during the Content Consumption operation.
- **Content Mediation Functional block (CMF):** Last but not least is the Content Mediation Function, which is considered to be the central function as "decision maker" in the CMP. It gets input from CRF, SNMF and PMF blocks, and it also interacts with Content Client during Content Consumption operation. It also interacts with the CAFF block in the CFP for the purpose of configuring delivery paths during the Content Consumption. Its main functionality is to make decisions regarding the selection of the best possible copy of the content, based on information about server and network conditions received from SNMF and the information about the available paths from the PMF, and then to setup and configure the *content delivery paths*. In particular, the CMF[2]:
  - receives content requests from the content client,
  - forwards the request to the CRF to be resolved,
  - gets information about all available content copies from the CRF,
  - gets long-term, offline info regarding the available underlying paths from the PMF,
  - gets up-to-date server and network condition information from the SNMF,
  - based on the above, makes decisions on both the most appropriate copy and on the most appropriate path, in order to save network and server resources and at the same time guarantee QoS to the content consumer,
  - and finally, setup and configure the selected path in CFP. Within COMET, this procedure is called *Path Configuration*[3] and is depicted with the arrow that points to the CAFF in Fig. 8.

Content Forwarding Plane

The CFP is presently composed of only one functional block, the ***Content-Aware Forwarding Functional block (CAFF)***. CAFF in the COMET system is a sophisticated forwarding function, which allows delivering content through paths selected in CMP (specifically by CMF). The CAFF will have enhanced capabilities in order to provide the required QoS for end-to-end content delivery. These capabilities are further discussed in D4.1 [3] and include traffic classification, point-to-multipoint forwarding, Network Address Translation (NAT) functionality in order to hide the *Content Server*'s IP address to the *Content Client*, etc.

---

[1] It must be noted that the SNMF functional block could be subject of evaluation in the refinement of the architecture expected for M36. Due to the different time scale required for network monitoring and server performance monitoring, it could make sense to separate those functions in different blocks.

[2] One of the functions of the CMF is the exposure of an interface to the Content Client for content consumption. As a result, the CMF block can be victim of attacks from malicious users. For this reason, it could make sense to move some of the functions from the CMF block to a different functional block. This will be a subject of evaluation in the refinement of the architecture expected for M36.

[3] It must be noted that the *Path Configuration* process could be subject of evaluation in the refinement of the architecture expected for M36. Its close relation to the network and the CAFF functional block makes it a candidate for a separate functional block as happens in other architectures (e.g. the PCRF in the NGN Control Architecture).

3.4 Decoupled and coupled approaches to map functional blocks into entities

After completing the high-level architecture, we map the functional blocks into a number of **entities** (Figure 9).

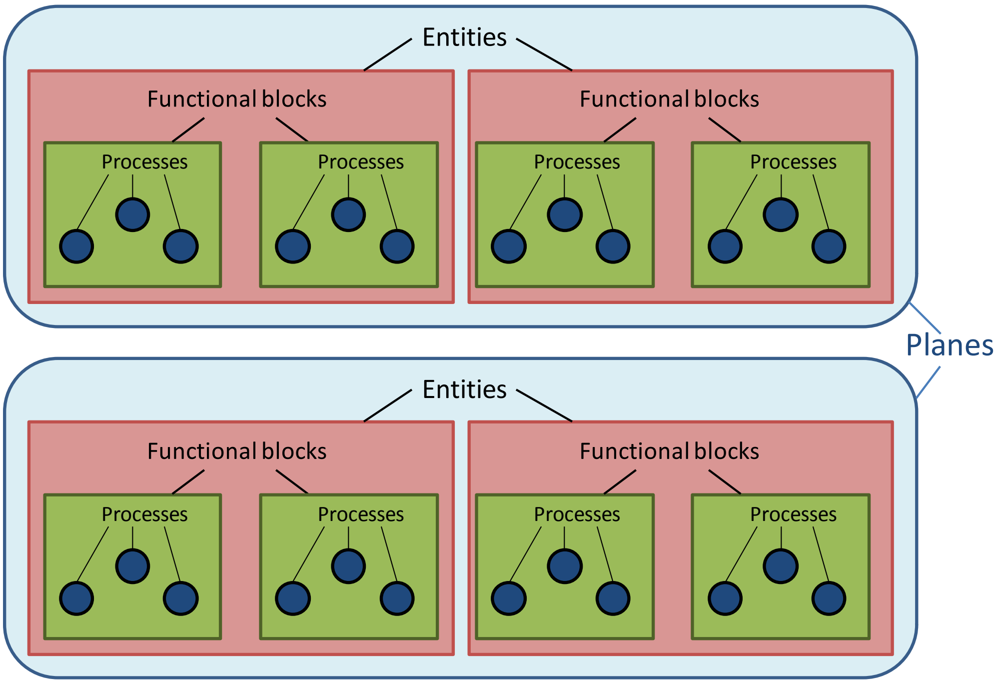

Figure 9: General mapping of functional blocks into entities

Entities must be understood as a collection of functional blocks, which interface with other entities to communicate each other.

The mapping of functional blocks into entities has led to two particular approaches in COMET, namely decoupled approach and coupled approach. These approaches differ not only in how functional blocks are mapped into entities, but also in the strategy to deploy entities, and they can be affected by potentially different scalability issues.

Based on the overall COMET architecture described in last section, two different approaches for mapping the functional blocks into entities have been designed and studied, namely decoupled approach and coupled approach, Both approaches differ not only in how functional blocks are mapped into entities, but also in the strategy to deploy these entities. Due to these differences, the two approaches may comprise different scalability properties.

The first approach (decoupled approach) is characterized by the following features:

- It follows the basic paradigm of the current Internet by allowing the physical signalling routes for content resolution and content delivery to be separated but coordinated.
- The content resolution sub-operation is based on a global directory system which stores content information, in a similar way as the current host-centric Internet uses global DNS directory system to support domain name resolution.
- There are specific entities that hold the CRF functional block and act as that global directory system for contents. These entities are different from the ones holding the CMF functional block.
- The entity hosting the CRF functional block is not linked to any specific network domain. On the other hand, the entity hosting the CMF is associated to a specific network domain.

For these reasons, the decoupled approach is considered an evolutionary graceful transition from the current host-centric Internet to the content-centric one.

On the other hand, the second approach (coupled approach) has the following characteristics:

- It follows a disruptive paradigm with respect to the current Internet, with the physical signalling routes for content resolution and the corresponding content delivery being coupled. More specifically, the *domain-level* content delivery path exactly follows the reverse direction of the original resolution path for each content consumption request.
- The content resolution sub-operation is performed on a hop-by-hop basis across intermediate domains.
- Content delivery paths in CFP are maintained with content states installed during the resolution phase in CMP.
- A unified entity holds both CRF and CMF blocks, and this entity is associated to a specific network domain.

The coupled approach breaks the current paradigm of resolution in the Internet. For this reason, this approach is considered a revolutionary scenario of content-centric Internet.

Just for the sake of clarification, in D2.1 [1] the terms Content Mediation Server (CMS) and Content Aware Forwarder were used to refer to the entities performing the processes currently covered respectively by the CMF and CAFF blocks. In next sections, these terms have been replaced to fit the current state of the architecture. The differences are as follows:

- the CMS has been replaced by the term Content Mediation Entity (CME) in the decoupled approach and by the term Content Resolution and Mediation Entity (CRME) in the coupled approach
- the CAF has been replaced by the term Content Aware Forwarding Entity (CAFE)

In this way, we homogenize the terminology by using the term entity, more appropriate for architecture descriptions, avoiding the use of the term “server” which usually refers to how entities are implemented in the end.

In the following sections, we provide an overview of these approaches based on the COMET architecture defined in the previous section.

3.4.1 Decoupled Content Mediation Approach

The decoupled approach follows the basic paradigm of the current Internet by allowing the physical signalling routes for content resolution and content delivery to be separated but coordinated. By decoupling the location of the content and the delivery, it is possible to have different architectures for both tasks, thus implementing the most appropriate one for each purpose.

This approach relies on the existence of a global directory system that stores *Content Records*, which are data structures containing content properties such as the QoS requirements for the content, the list of available content sources or the application protocols to be used to retrieve the content from each *Content Server*. This global directory system resolves from a content name to a *Content Record*, using for that resolution a hierarchical architecture similar to the one used in DNS.

In this approach, the functional blocks described in Section 3.3 are mapped into the following entities:

- Content Resolution Entity (CRE): This is the entity which encompasses the function of CRF. This function consists in keeping track of the content records or indexes and in resolving Content Names to Content Properties.
- Content Mediation Entity (CME): This is the entity which encompasses the Content Mediation Function (CMF) and is the one which the content client contacts first in order to

consume a content. Apart from requesting the Content Record to the CREs (name resolution) and obtaining the paths from the Content Servers to the Content Clients (path discovery), the main function of this entity consists in deciding the best server and delivery path based on the content characteristics and on the awareness provided by other entities (RAE, SNME, CRE). Finally, this entity is also responsible of configuring the content delivery paths.

- Routing Awareness Entity (RAE): This is the entity that encompasses the Path Management Function (PMF). This entity gathers Network Reachability Information and provides it to the CME, in a proactive offline manner, in order to obtain the necessary routing awareness to perform the path discovery process and make decisions.
- [redacted]
- Content-Aware Forwarder Entity (CAFE): This entity encompasses the Content-Aware Forwarding Function (CAFF). This function consists in delivering the content through paths instructed by the CME.
- Content Client: This entity interacts with the CME to request contents through a unified interface (the same for all Content Clients).

[redacted]

- Content Publisher: This entity interacts with the CRE to perform the Content Registration through a unified interface (the same for all Content Publishers).

As will be shown in section 3.4.2, it must be noted that the RAE entity is the same in both approaches. This entity should be implemented in each domain and is in charge of gathering network reachability information across domains from other remote RAEs and providing routing awareness to the entity encompassing the CMF functionality (i.e. CME in the decoupled approach and CRME in the coupled approach). RAEs can be understood as inter-domain routing entities similar to BGP speakers that perform the off-line routing awareness process in long time scale. Based on the inter-domain provisioning information from the inter-domain SLA agreements and the intra-domain provisioning information, each RAE exchanges its Network Layer Reachability Information with other RAEs in peering domains to build inter-domain routing paths. Nevertheless, two main features make RAEs differ from BGP speakers:

- Network Layer Reachability information is exchanged in terms of COMET Classes of Service that are globally known by COMET-enabled ISP networks,
- RAEs are able to propagate information about a number of alternative paths in addition to the default one in order to offer differentiated content delivery across domains

More details about RAEs and the routing awareness process are provided in D4.1 [3].

The following figure graphically shows the mapping of the functionalities to the entities and the interactions between them:

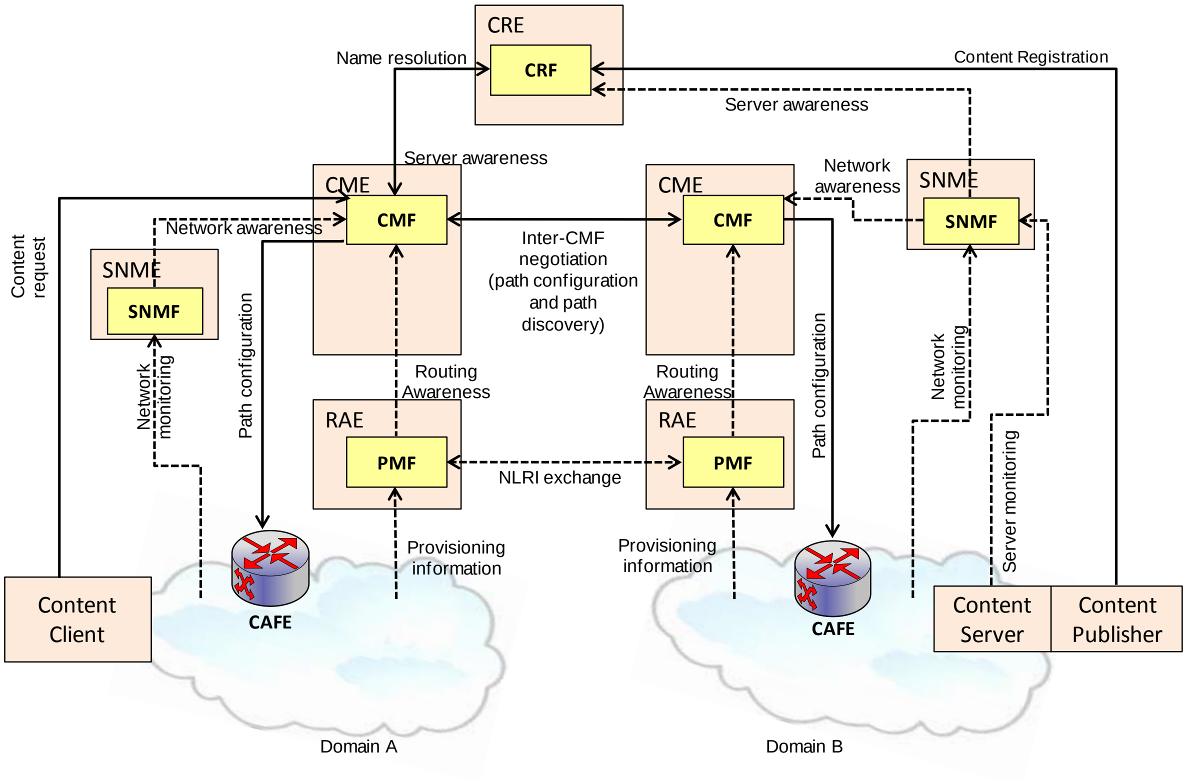

Figure 10. COMET Decoupled Content Mediation approach – Architecture and Information Flow

As mentioned earlier the decoupling of content resolution and delivery allows applying different architectures for each purpose. Thus, CREs, in charge of storing the content records and providing name resolution, are not associated to any particular network domain. On the contrary, CMEs and RAEs are associated to network domains.

Now we describe the specific content-based operations within the COMET system and the entities and functionalities involved in these operations. Fundamentally, there are two main operations: Content Publication and Content Consumption. The latter operation can be further divided into two sub-operations performed per content request: Content Resolution and Content Delivery. Below, we explain how these basic operations and sub-operations are dealt in the COMET System.

Content Publication

We consider Content Publication as the process of making content available to Content Consumers. More specifically, the Content Registration is the one-step process inside the Content Publication which deals with the registration of a specific content in the global directory system built by CREs (creating or updating the associated content record). The content registration involves two entities: the Content Publisher and the Content Resolution Entity.

Figure 11 shows the entities and functions involved in the Content Registration of the Decoupled approach for the COMET Architecture:

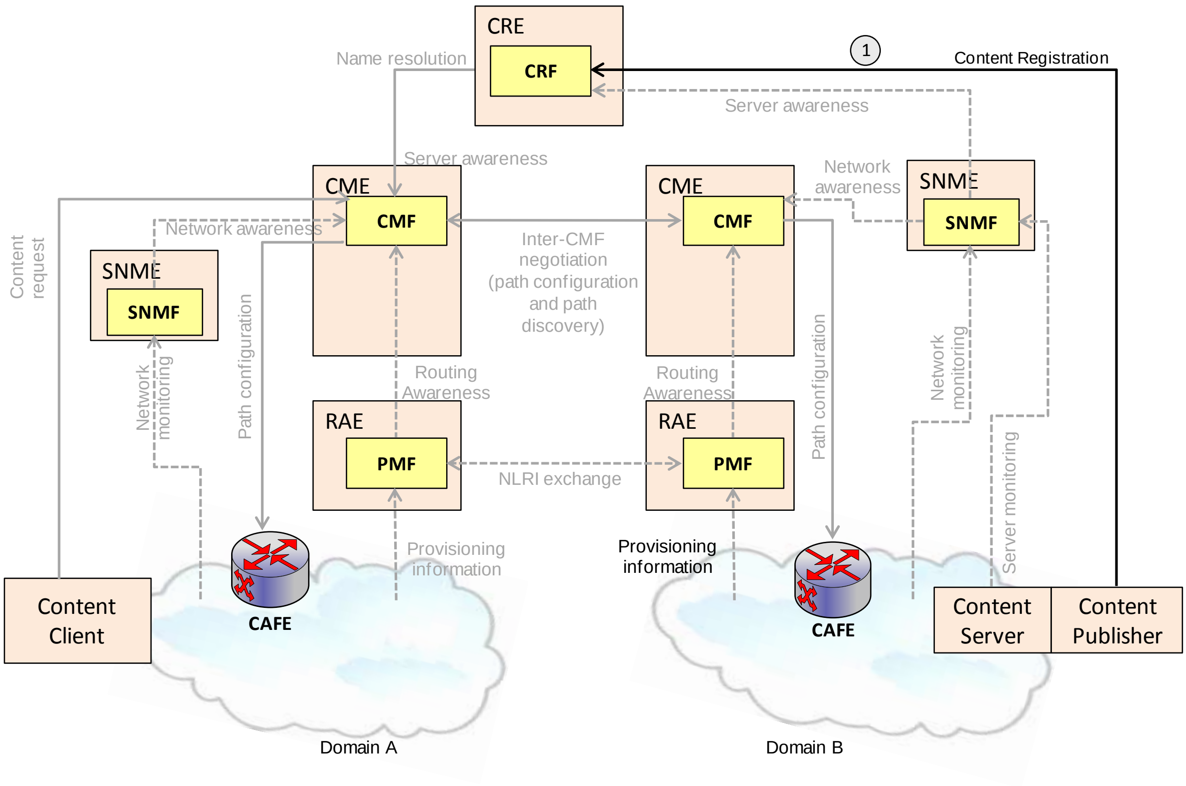

Figure 11. Content Registration according to the Decoupled Content Mediation approach

In this process, the Content Publishers must notify the COMET System that there is a content that they want to make available for access across the Internet. The Content Publisher will also specify the Content Name associated to that content. This is accomplished through a unified COMET publication interface accessible for all Content Publishers. As part of this process, some parameters associated to the content itself are passed to the CRE by the Content Publishers. Some examples of these parameters are the QoS requirements of the content, the location of the content copies and its way of distribution (transport protocol, application protocol, etc.). With that information, the CRE creates a Content Record including these properties associated to the content that is being published.

After the creation of the Content Record, a unique Content Identifier could be returned from the Content Resolution Entity to the Content Publishers.

Once the content has been registered in the COMET system, the CMEs are responsible for making the content properties available throughout the Internet so this content can be reachable by all Content Clients of the different network domains.

Content Resolution

We consider Content resolution as the sub-operation responsible for the discovery of the requested content based on a given Content ID or Content Name. Particularly, in the Content Record-based resolution approach, all the information related to the content, servers and its characteristics is stored in Content Records, which are located in the CREs. These Content Records are reachable either through the Content Identifiers or through the Content Names, which must be known before hand by the Content Clients in order to access that content. The sub-operation of Content Resolution has the following steps:

1. The whole process is triggered right after the Content Client requests a content to the Content Mediator Entity (CME). In this step, the Content Client sends a Content Name or a Content ID in order to express his purpose of retrieving a particular content.
2. After the request, the CME should be able to locate all the copies of the same content if the content has been replicated and hosted at various content servers and the paths to reach

those particular servers. This is crucial for optimization/enhancement of the content delivery and also, enabling capabilities such as anycast. To achieve this, the CME starts the Name Resolution process which consists of sending the Content Name or Content Identifier to the appropriate CRE in order to obtain the Content Record associated to the content.

As it has been detailed in the Content Publication section, there is a Content Record for each content which has all the information associated to this content. This Content Record contains the Content Properties that will be later returned to the Content Client in order to make him possible to retrieve the content from the Content Server.

3. With the information included in the Content Record about the available content sources and the one that has been collected from the RAE about the Network Reachability Information, the CME determines/discovers the set of available paths from the Content Servers to the Content Clients. This process could require an interaction with CMEs in other network domains (see D3.1 [2] for more details).
4. Then the CME makes the decision of the server and path that will be used to deliver the content from the Content Server to the Content Client based on the requirements of the content, the QoS long-term parameters of the path, the server conditions and the network conditions.
5. When the CME obtains the selected duple server/path, it configures the path in order to prepare the CFP for the delivery. The CME interacts with CMEs in other network domains and with the CAFEs inside its network domain for this purpose.
6. Finally, when the path is configured, the set of content properties necessary to invoke the appropriate application and get the content from the Content Server are returned to the Content Client. It must be noted that these properties are a subset of all data contained in the Content Record.

Figure 12 shows the entities, functions and processes involved in the Content Resolution of the Decoupled approach for the COMET Architecture.

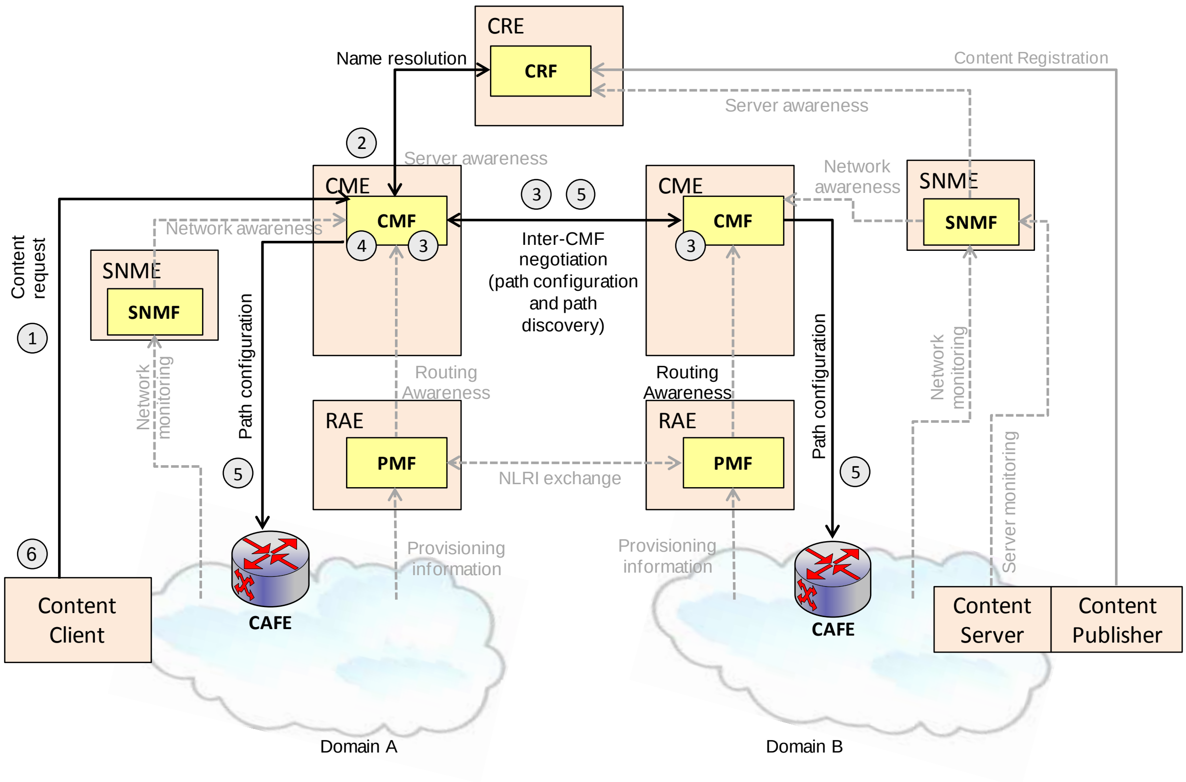

Figure 12. Content Resolution according to the Decoupled Content Mediation approach

Content Delivery

The content delivery sub-operation consists in the transfer of the content from the Content Server to the Content Client. It involves three different entities of the COMET architecture: the Content Client, the Content Server and the CAFEs. CAFEs are instructed during the content resolution on how to transport the content. These instructions include how to classify the content packets and how to forward the packets through a specific path of CAFEs. The detailed description on the content delivery operations corresponding to this part of the COMET architecture is described in D4.1 [3].

It must be noted that not all network routers are required to be CAFEs. We envision that the routers that will support the CAFF functionality are the ones that reside at the edges of the domain.

3.4.2 Coupled Content Mediation Approach

The current Internet relies on the DNS system whereby a request is usually resolved first by querying the DNS system before the actual request is being sent to the resolved content server that hosts the requested content. The main strategy of this coupled approach is to simplify the process by combining the two round-trips into one single operation. Essentially, the resolution procedure is coupled with the content delivery procedure and thus saving one round-trip. This requires more radical changes to the intrinsic working of the current DNS-IP based Internet structure. In this section, we outline how the COMET architecture described earlier can achieve the coupling of these processes.

As discussed previously, the functional blocks described in Section 3.3 can be grouped into either one or more entities. These entities will represent the actual implementation of COMET functions into physical machines.

In this approach, these functions are grouped into two physical entities.

- *Content Resolution and Mediation Entity* (CRME) – this entity, typically owned by individual ISPs, encompasses the CRF, CMF and SNMF functional blocks from the COMET architecture.
- *Routing Awareness Entity (RAE)* – this entity encompasses on the PMF functional block. It is the same corresponding entity in the decoupled approach (cf. Section 3.4.1) where the Network Reachability Information is obtained and fed to the CRME (more specifically, to the CMF block).

We note that according to this approach, each COMET-enabled domain has to have (at least) one CRME implemented or having an agreement with one of its neighbouring domains for the access of the neighbour CRME . The CRME interfaces with all internal and external COMET entities, in order to accomplish the publication, resolution and delivery operations. In addition, each CRME interfaces with other CRMEs, located in the same or other domains, resulting in a network of CRMEs, which constitute the CMP.

We detail the *information flow* between CRME and non-CRME functions/entities, as well as the inter-CRME function relation. The *communication flow* between the different entities of the system is the subject of deliverables D3.1 [2] and D4.1 [3].

We show in Fig. 13 the coupled approach under the COMET architecture, as well as the *information flow* between the different functions and entities. This figure includes all the required operations and functional component interactions from content publication to the COMET system, to content resolution and finally to path-setup for the content delivery.

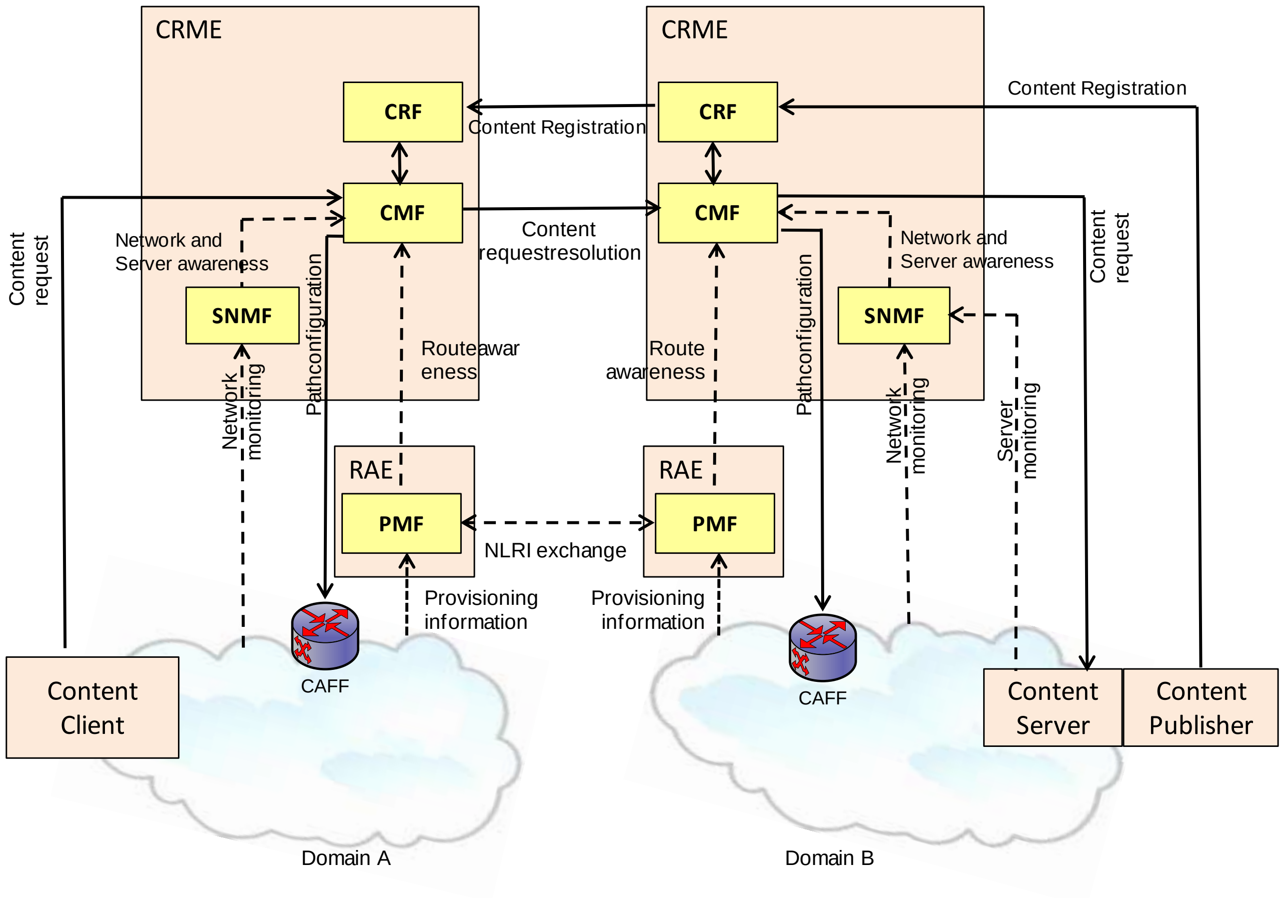

Figure 13. COMET Coupled Content Mediation approach – Architecture and Information Flow

From the figure, we illustrate the simplest scenario where there are only two neighboring COMET-enabled domains with each having a single CRME respectively. In reality, the CRMEs may be connected to many peer, customer and / or provider domain(s). As described previously, the networking and server monitoring along with route awareness are done periodically. The publication and resolution processes, however, are triggered per content request.

We break the above figure in two, according to the COMET operations discussed in the beginning of this section, namely the *content publication* (Fig. 14) and *content consumption* (Fig. 15). *Content consumption* is further divided into *content resolution* and *content delivery*. In Section 3.2, we discussed the responsibilities of each of the functions of the CMP block and the CFP block. Here, we discuss:

i) how the coupled content mediation approach implements the sequence of events,

ii) how it handles the different functional blocks described before and

iii) the interactions between different entities of the system.

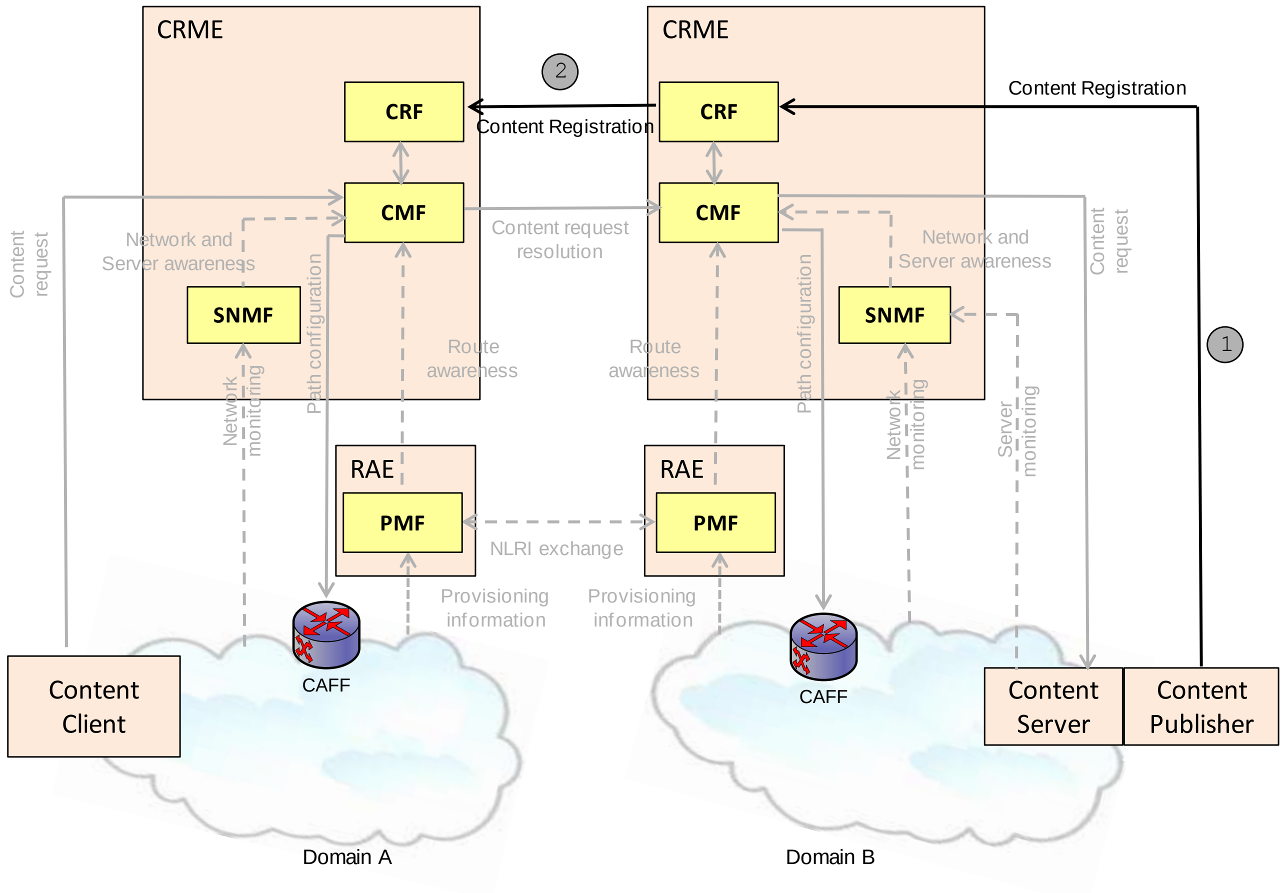

Figure 14. Content Publication according to the Coupled Content Mediation approach

Content Publication

To enable the coupling of the resolution and delivery processes, the way a piece of content is published for the consumption in the Internet has to prepare the foundation that "guides" the resolution. The general idea is to allow the business relationships (peer / customer / provider) amongst the domains (ISP networks) to dictate the publication path across domains. This is compatible with the current BGP routes, which are also configured according to inter-domain business relationships.

The first step of the content publication is done locally between the content publisher and the local Content Resolution Functional block (CRF) of the local CRME (step 1 in Fig. 14). Basically, after uploading the content to the hosting content server, the content publisher issues a publish message to its local CRME (or immediate delegated CRME in the case where there is no local CRME). Upon reception of this message, the local CRME (specifically its CRF functional block) will create a new content record for this content.

As a second step, the CRF block in the local CRME has to inform its counterpart in other CRMEs of the COMET system of the existence of this new content (step 2 in Fig. 14). However, the propagation of this new content record is not done in a broadcast manner where all CRMEs in the entire Internet will know about the new content. This is due to scalability considerations. Rather, the approach defines specific publication rules that propagate the record to the ones that need to be informed, in order for all contents in the COMET system to be accessible from all content clients. The specifications of this process are described in detail in D3.1 [2].

Content Resolution

We further illustrate the sequence of events for the *content resolution* part of the *content consumption* in Fig. 15 which consists of three distinct steps:

1) The content client sends its content request to the CMF block of the local CRME. During the *content resolution* process, the CMF consults the CRF (belonging to the same CRME) to resolve the received request. It will first check if the CRF block has the record of the requested content. Based on the result of this query, the CMF block will decide on the next resolution step.

2) The next step consists of two separate processes.

    i. The first process continues the resolution sub-operation. It is dependent on the answer obtained from the CRF. If a positive result is returned from the CRF, then either the implicit or the explicit location of the content is known. That is the local CRF block at the content client's domain knows either the exact location of the content, or the path to follow in order to find the exact location. The CMF can then follow this information to find the targeted content server. On the other hand, if a negative result is returned from the CRF block, the CMF follows specific rules designed for this approach to the further discovery of the content. Detailed procedures will be described in D3.1 [2].

    ii. The second process relates to the construction of the content delivery path in the local client's domain (i.e., Domain A in Fig. 15). This is basically the preparatory stage that is required to couple the resolution and delivery processes. The CMF block gathers information about the available paths of the local domain (Domain A in Fig. 15) from the PMF, and optionally the SNMF blocks, as discussed previously, and applies the required *Path Configuration* to the underlying paths, or in other words, prepares the paths for the *content delivery in the client's domain*.

    This step is repeated until the request reaches the domain that hosts the content requested.

3) In this final step, the CRME of the domain where the content is actually located knows the explicit location of the content server. Similar to the previous step, it has to *enforce the path*, but this time towards the content server itself. Then, it forwards the content request to the corresponding content server to initiate the transmission of the content. The content transfer itself takes place at the lower content forwarding plane. The specifics of these operations are detailed in D4.1 [3].

We note that in the above scenario (and in general in case the content client and the content server reside in different domains) the *Server and Network Monitoring Functional block* (SNMF) at the client's domain (Domain A, in Fig. 15) provides *Network* information only, while at the server's domain (Domain B, in Fig. 15) provides both *Network and Server* monitoring information.

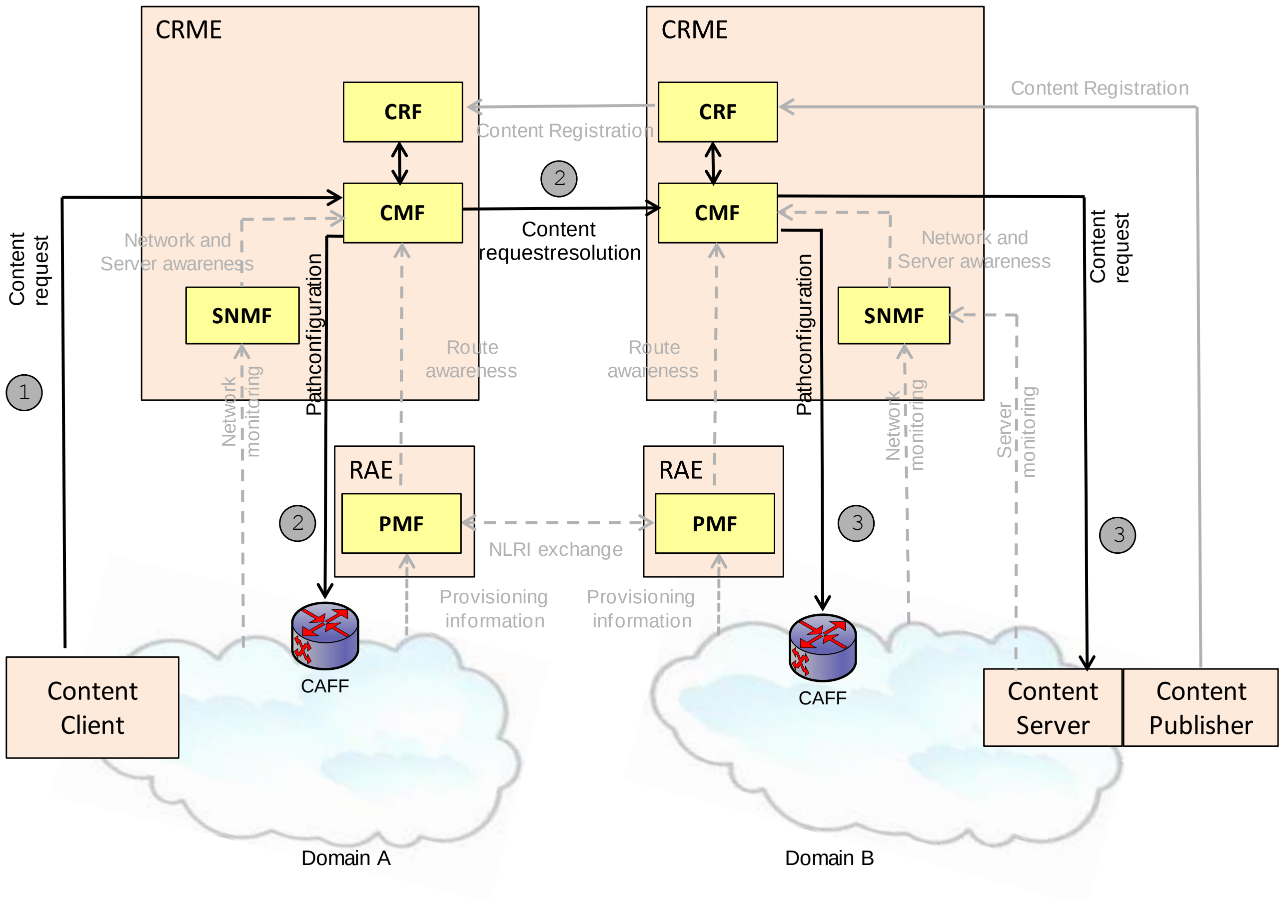

Figure 15. Content Resolution according to the Coupled Content Mediation Approach

Content Delivery

The content delivery part of the content consumption involves the actual network and in particular the *Content-Aware Forwarding Functional block* (CAFF). The CAFF block will be implemented in the *Content-Aware Forwarding Entity* (CAFE). Basically, CAFEs forms the CFP.

The CAFE includes all the required mechanisms that guarantee smooth delivery of content back to the content client. The delivery path configuration mentioned in the previous section essentially means that the CMF install specific content states on the relevant CAFEs within its own domain regarding the specific content request. The detailed description on the content delivery operations corresponding to this part of the COMET architecture is described in D4.1 [3]. Here, we note that we do not require all network routers to be CAFEs, or to be implemented as CAFF blocks. We envision that the routers that are going to be enhanced with CAFF functionality are the ones that reside at the edges of the domain.

3.5 Borderline between WP3 and WP4

In this section, we attempt to draw the line between WP3: *Content Mediation System* and WP4: *Content-Aware Network Enhancement*. The content of the corresponding Deliverables D3.1 [2] and D4.1 [3] is split accordingly.

We describe the split between the two WPs first in terms of the *COMET operations*, and then in terms of *functional blocks* as these were defined and described before. We provide the overall system architecture again in Fig. 16 below for clarity.

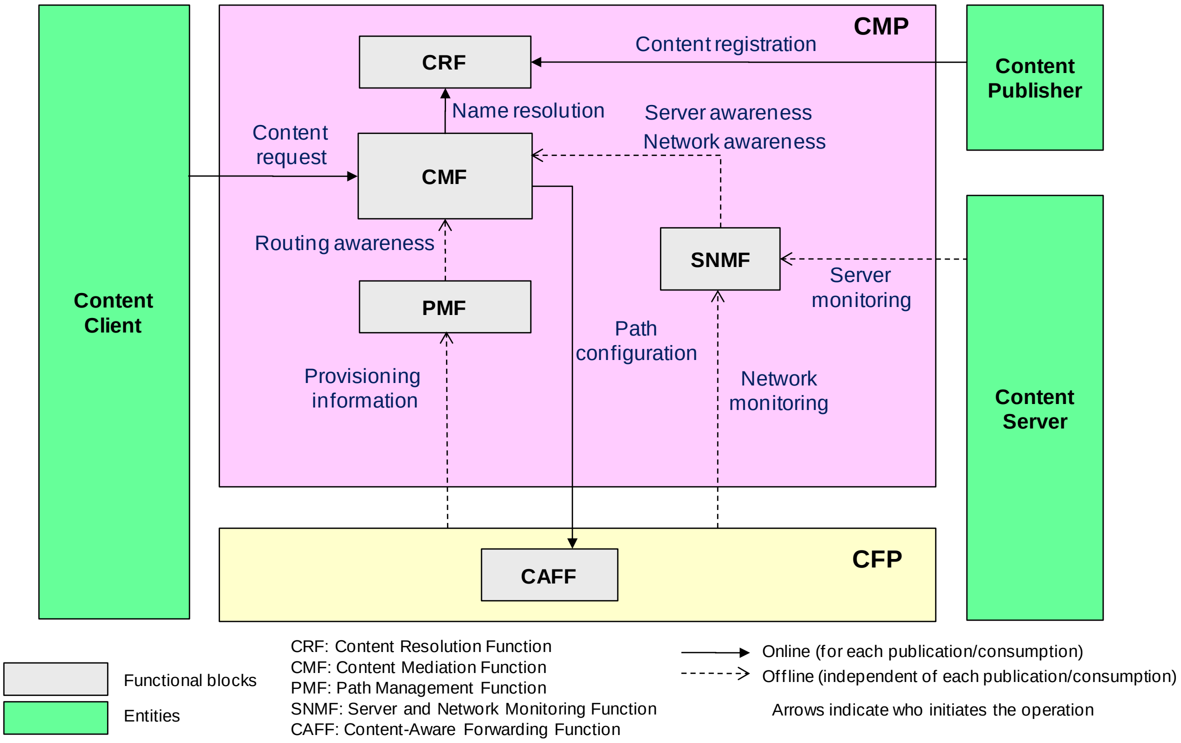

Figure 16. Overall COMET Architecture

In terms of *COMET Operations*, it is clear that the *Content Publication* is part of WP3, while the *Content Consumption* operation is split between the two WPs. From the processes inside the *Content Consumption* operation, we consider that the processes of providing server and network awareness, name resolution, path discovery and decision making are part of WP3 and will therefore be explained in detail in D3.1 [2], while the rest of processes (providing offline routing awareness, path configuration and content-aware forwarding) are part of WP4 and will be the subject of D4.1 [3].

Regarding the functional blocks, after the definition of the overall COMET Architecture it is pretty straightforward to draw the borderline between WP3 and WP4. On the one hand, most of the functional blocks that belong to the *Content Mediation Plane* (CMP), namely the *Content Resolution Functional block (CRF), Content Mediation Functional block (CMF),* and the *Server and Network Monitoring Functional block (SNMF)* belong to WP3.

On the other hand, the *Content-Aware Forwarding Functional block (CAFF)* that belongs to the *Content Forwarding Plane (CFP)* is the subject of investigation for WP4. Regarding the *Path Management Functional block (PMF)*, although it is part of the CMP, its main responsibility is providing routing awareness and, since this process is closely related to the transfer capabilities offered by the CAFEs in the CFP, we considered more appropriate to describe it in D4.1 [3].

It is clear, however, from Fig. 16, that some functional blocks of the CMP interact with elements of the CFP. Namely, the CMF block *configures* the path for the content delivery; the SNMF block gets some information regarding the state of the network (i.e., network-awareness); and the PMF performs routing awareness and network provisioning, both of which require interaction with the CFP. Therefore, these processes belong partially to WP3 and partially to WP4, depending on the specific mechanism that is investigated or defined every time.

4 Mapping of the Use-Cases on the COMET Architecture

This chapter describes the mapping of the use cases that were introduced in *D2.1 Business Models and System Requirements for the COMET System* on the COMET Architecture defined in the previous chapters of this document. These use cases have driven the design of the COMET architecture and, accordingly, will drive the future demonstration activities.

As it was introduced in D2.1 [1], these use cases refer to the distribution of contents and services in which there exists mediation in the network. Given that Network Operators implementing the COMET system will be content-aware, new and different scenarios are possible regarding content distribution through the Internet.

Next sections give a detailed description of the mapping for the different use cases that have been considered as part of the COMET project.

In all cases, it must be highlighted that the COMET architecture is still under design and, therefore, some details could change in the course of the project, although the main concepts of the mapping of the use cases will remain valid.

4.1 Use case 1: Adaptable and efficient content distribution

4.1.1 Brief description

This use case presents a scenario where a particular Content Provider *dynamically and transparently adapts the global capacity of its servers* in order to cope with the required demand for the distribution of a live event through the Internet. This adaptation is made while the content is being served, which proves to be very useful as the audience of live events is usually unexpected and difficult to predict.

For the delivery of the content, this scenario contemplates the existence of multicast capabilities globally available in the ISP network, in much the same way as it happens today with other operator services (e.g. IPTV). With the use of these mechanisms, the traffic is distributed in a more efficient fashion, optimizing the use of resources and minimizing transmission costs in the local ISP.

In addition, the ISP is able to offer QoS capabilities to the Content Provider in order to deliver its contents in a guaranteed way. With this achievement, the content delivery is transparently associated to the most appropriate class of service/network plane, which ensures the users receive the QoE they expect and that the network is used as efficiently as possible.

Furthermore, once the live broadcast has finished, the Content Provider makes a recorded version of the same event available for the end-users. Now, the VoD content can be retrieved using the same identifier as the live content despite the distribution mode change (i.e., from live to recorded). So, the identifier is always the same regardless of whether the event is live or recorded, thus simplifying the clients' access to the content.

4.1.2 Storyline

The storyline of the mapping of this use case is as follows:

1. The TV channel, acting as the Content Publisher, prepares for the live broadcasting of the football match. For that purpose, the channel:
    a. sends the content information to the CRF Block (Content Resolution Functional Block) for the Content Registration. This information includes details of the content stream and associated metadata (e.g., number and type of sub-streams, QoS requirements, the set of live streaming servers that will be supplying the content,

available distribution modes etc.). These are largely the content characteristics, or content properties referred to in the previous sections.

   b. obtains a unique identifier (i.e., Content ID) and, optionally, a human-readable identifier for the match (i.e., Content Name). The Content Name can potentially be specified by the publisher himself.

2. The TV channel starts to broadcast the live content from its Content Servers.
3. End users launch applications in their end devices to request the live broadcast by:
   a. obtaining the unique Content ID or Content Name via either a search engine, electronic programme guide or another out-of-band mechanism (e.g., link in a blog, word of mouth, newspaper article, etc.)
   b. requesting the video stream from the CMF Block (Content Mediation Functional Block) by sending the unique content ID, or Content Name to this functional block as part of the Content Request operation.
4. The CMF Block resolves the request, chooses the server and path to be used, as part of the decision process, and enforces the establishment of this communication path according to the requirements of the content (QoS, resilience, cost, etc.).
   a. This includes retrieving the Content Record with all the information associated to the content from the CRF Block based on the Content ID (or Content Name). Next, the CMF Block chooses the optimal streaming server (anycast mode) based on the metrics (e.g., server performance) gathered from the SNMF Block. For instance, it can bind it to the closest streaming server, judging in this case by the number of hops to the server.
   b. Later, the CMF selects the optimal path for the content delivery from the chosen server to the originating user. For this decision, the CMF needs to take into account the QoS, cost, resilience, etc. requirements for the particular content in question (which are present in the Content Record) and the information gathered from the PMF Block regarding the network reachability information. In this use-case, for instance, we are dealing with live content; hence low latency is of utmost importance.
   c. If available in the local ISP, a multicast group can be used in order to optimize network resource usage. This enhancement would be enforced by the CMF in the entities that will be in charge of the CAFF blocks.
5. During the live broadcast, in case more capacity is needed, the TV channel can transparently add new live streaming content servers. This can be done by either updating the Content Record (associated to the unique identifier) by sending the new locations to the CRF, or by sending a new publication request to the CRF Block to include the new content locations. In that case, the initial content record will have to be deleted, but the old entry (i.e., record) will still need to be associated with the new one.
6. Once the live broadcast has finished, the TV channel (as the Content Publisher) updates again the Content Record in the CRF to associate the Content ID (or Content Name) to the recorded version of the same event. This update will include the details of the location of the new VoD servers and the associated content characteristics to show, for instance that this is non-live content, so multicast is no more available; furthermore, some QoS requirements may be relaxed for the same reason (i.e., recorder versus live).
7. End users can now connect to the VoD version of the content making a request to their associated CMF Block using the same unique identifier. This step is analogous to the case of live content with the difference that the number of sub-flows might be different. The Content Aware Forwarding Function (CAFF) is now instructed to associate the flow to a less restrictive class of service, and multicast capabilities of the local ISP can no longer be used.

8. In case of a reduction in the number of content requests, the TV channel might reduce the number of VoD content servers. The Content Name and the Content ID will remain the same, however, so whatever the changes in the location or number of servers will be handled by the Content Publisher and the CRF Block and will be transparent to the clients.

4.2 Use case 2: Handover of content delivery path in a multi-homing scenario

4.2.1 Brief description

This use case focuses on multi-homed users, who inherently have the option to choose between more than one interfaces and therefore paths to connect to the network. Two hosts (i.e., either servers or end-users) can choose among more than one path to exchange traffic between them, as long as one of them is multi-homed. Therefore, the need for multipath support at the network level is eliminated. Here, we focus on the case where, users can indeed switch between different paths, based on the respective network conditions, but without the support of multipath routing protocols implemented into the network.

More precisely, we focus on the case, where a content consumer, during an ongoing session, wants to make a request for a handover to the second network/ISP that he is connected to. The most common and obvious reason for such a handover between two different interfaces/ISPs is quality of experience.

4.2.2 Storyline

As described above, according to our assumptions herein, there is no need for multipath routing protocol support, in order for the content client to be able to switch between his two interfaces. Next, we describe the steps, according to which the end-user requests from the COMET system handover to the second interface. Our description is based on the COMET Architecture provided earlier in Chapter 3.

1. Initially, the content consumer is receiving VoD content from one of its interfaces with a dedicated IP address.
2. Assuming that the consumer wants to switch to his second interface, it makes a request for a handover to the local CMF block of the ISP that he wants to switch to, i.e., not to the one that is already connected to. (The opposite is possible too, but this would change the whole flow in the rest of the steps below accordingly.)
3. Upon receiving such a request, the CMF block of the new ISP will have to make a decision on whether such a handover is desirable or possible. This decision will be based on:
    a. the preferences from the content consumer, which are included in the handover request,
    b. the overall network conditions in the underlying paths. The CMF block gathers this information from the SNMF block, as described earlier.
4. Once the CMF block decides to accept the handover, it consults the PMF and configures the underlying path through *Path Configuration*.
5. Only after the new path is setup and enforced, will the old connection be dropped. This will ensure seamless handover to the content consumer, whose application will resume the consumption of the content through the new interface.

4.3 Use case 3: Webinar "All about CDNs"

4.3.1 Brief description

This use case presents a scenario where a *Webinar* is created in order to deliver contents from a presenter to multiple viewers with specific QoS requirements. This is done with the help of a *webinar* platform that allows offering live *webinars* and also the possibility of recording the *webinar* so that it can be saved and cached on edge servers.

In a *webinar* scenario there are usually two main roles: the presenter and the listener. These roles, which are assigned to the specific users, are not static and can switch at any time during the session, requiring changes in the path configuration. Along the description, the user who initiates the *webinar* and leads it is usually referred to as the initial presenter.

In order to simplify the mapping of this use case to the functional architecture of COMET, we assume that there exists just one central server in the scenario. This server can be identified as both Content Publisher and Content Provider. The initial presenter and the listener both represent the entity of Content Client.

For this particular scenario, it is required that the Content Resolution Functional block must be able to deal with Content IDs and meta-Content IDs. Therefore, it must provide the capability of registering the complete *webinar* as a meta-Content and obtain the associated meta-Content ID. This ID is subsequently used by the *webinar* server to register single contents and associate them with a particular *webinar* (using the meta-Content ID). When the *webinar* server registers new streams, it also has to indicate the required QoS for each of them.

4.3.2 Storyline

The storyline of the mapping of this use case is as follows:

1. The user-presenter, acting as the Content Provider contacts with the *webinar* server in order to create a *webinar* prior to the meeting.
2. Then, the *webinar* server, acting as the Content Publisher, contacts with the CRF in order to publish the *webinar*, specifying its time and duration, as well as its subject. Specifically, the *webinar* server registers in the CRF each single content related to the *webinar* (audio, video, chat, etc.) with:
    - single content metadata describing its particular properties (e.g. bitrate, quality, etc.)
    - location of the server (IP address)
    - QoS requirements for the content
3. The *webinar* server obtains one unique content ID for each content registered this way.
4. These content IDs are merged into meta-Content IDs by the *webinar* server, which will contact the CRF to create this merging. As a result, two meta-Content IDs will be generated: one for the presenter and a different one for the attendees. It must be noted that a same content ID can be associated to different meta-Content IDs. For instance, the chat is expected to be the same for the presenter and the attendees, so that it could be possible to associate the presenter's content ID of the chat to the attendees' meta-Content ID.
5. The initial presenter starts the *webinar* procedure. With the unique meta-Content ID associated to its role obtained previously (for instance by using a search engine), the presenter performs the Content Resolution by asking to the CMF about this content. This request is received by the Content Mediation Functional block located in its domain.
6. The presenter obtains the list of single ContentIDs of the different streams. For each of them (audio, video, text), it performs the Content Resolution operation.

7. When the CMF receives the presenter's request, it finds the location of the Content Server ("origin server") and tries to configure the communication path between the presenter and this server according to the specific content requirements. When the communication path has been established, the *webinar* is able to start.
8. The Content Clients that want to join the *webinar* session must perform the Content Resolution operation sending the meta-Content ID of the *webinar* to the CMF of their domain. This identifier must have been previously obtained by using some search engine or via published invitations.
9. The CMF returns to the Content Client the list of single Content IDs for each of the contents inside the *webinar*.
10. Each Content Client performs the operation of Content Resolution for each of the contents that compose the *webinar*.
11. The CMF, after every request, identifies the origin server and triggers the configuration of the communication path according to the stream requirements and the awareness provided by the rest of functional blocks of the architecture.
12. Once all paths for each content are established, the Content Client will be able to receive the different contents of the *webinar*.

This storyline is independent on whether a decoupled or a coupled approach is followed. However, it must be noted that the first resolution step (from meta-Content ID to a set of content IDs) could not be done with the hop-by-hop resolution approach.

All access control is done on the *webinar* server and is outside the COMET scope. If the user is not allowed to present, the session will not be accepted.

4.4 Use case 4: P2P offloading

4.4.1 Brief description

This use case presents a scenario where a Content Provider wants to distribute a live event through the Internet, using a traditional client-server model. However, when the content servers are fully busy, being unable to serve the streaming to additional clients, the provider would want to dynamically offer the possibility of serving the content via P2P streaming, so that the content remains accessible to all potential viewers. In such a situation, new clients will launch their P2P streaming applications to access the content.

The change in the way of distribution would be made without disruption to the previously connected users and transparently for the end user thanks to the use of the same Content Name (or Content ID) for all modes of distribution. Hence, this name or identifier is agnostic of whether the type of distribution is based on client-server model or on P2P streaming.

Such a desirable scenario will benefit Content Providers (from the most popular ones to the small Content Creators) which now can deliver their contents to all the audience even when they do not have enough bandwidth or server capacity to serve them through a client-server model.

4.4.2 Storyline

The storyline of the mapping of this use case is as follows:

1. The Content Provider prepares for the live broadcasting of the event. It will prepare two kinds of Content Servers (normal streaming servers and P2P streaming sources) and will perform the Content registration operation by sending the Content information and characteristics to the CRF Block. For that purpose, the Content Provider will necessarily be a Content Publisher to interact with the COMET system. The steps are the following:

   a. Registers the content in the CRF Block, by sending the information associated to the Content. That is, for instance, the two ways of distribution of the content, the details for each way of distribution (set of sources, protocols, MIME types, etc.) and the conditions under which to switch between them, pointing out that P2P streaming should be used only in case of server or path congestion. Optionally, the TV channel specifies a Content Name for the content for its registration.
   b. Obtains one unique global identifier for the content (Content Name or Content ID), valid for both ways of distribution. Additionally, it will be necessary a mapping between the Content Name (or Content ID) and a content identifier in the particular P2P platform. The Content Resolution Functional Block, or some servers in the P2P platform must be aware of this mapping, although the specific details are still to be defined.
2. End users use their Content Clients to request the live broadcast. The Content Client obtains the unique Content Name or, optionally, the Content ID via search engine, electronic program guide, or out-of-band mechanisms (e.g. word of mouth).
3. The request is sent to the particular CMF Block associated to the Content Client by using the Content ID or the Content Name. The CMF will later return to the client the necessary parameters in order to launch the appropriate application and retrieve the content.
   a. If needed, the Content Mediation Functional Block prepares the Content Aware Forwarding Function (CAFF) accordingly. This can be achieved by enhancing the network layer with the necessary rules for the delivery of the content.
4. Initially, the interested content clients will retrieve the content stream directly from the servers following the traditional client/server paradigm.
5. Once the uplink capacity of the Content Servers is saturated, the CMF Block will be informed by the SNMF Block and will dynamically switch the mode of distribution to P2P streaming by updating the Content Records in the CRF Block. Here we assume that:
   a. The SNMF Block gathers the Server Condition information for the Content Servers that are providing the content and informs the CMF Block in order for it to act accordingly.
   b. Therefore, in case of server congestion in terms of load or bandwidth, the CMF Block will become aware and will change the distribution mode to P2P streaming; this will be coordinated in cooperation with the CRF Block.
6. Content clients trying to consume the content with the same Content ID or Content Name will be provided with the parameters from the CMF Block to setup their P2P streaming applications accordingly to retrieve the content.
   a. The video stream is requested from the CMF Block associated to the client by using the same Content ID/Content Name.
   b. Again, if needed, the CMF Block prepares the Content Aware Forwarding Function (CAFF) accordingly. This is again achieved by enhancing the network layer with the desired rules for the delivery of the content.
7. The content is distributed now via the overlay network built by the P2P streaming application:
   a. The Content Server provided by the Content Provider acting as source node is connected to the P2P overlay network as one of the seeds of the content.
   b. All new Content Clients are now connected to that P2P streaming overlay network and can act as distributors of the content (as in any other P2P distribution platform).

5 Coverage of System Requirements in the COMET Architecture

In D2.1 [1], we defined a number of system requirements that the COMET system has to fulfill. In this chapter, after defining the *High-level Architecture of the COMET System* in the previous ones, we detail which and how these requirements are covered by the COMET system architecture. We go through the requirements defined in D2.1 [1] one by one. For completeness, we give the original system requirement in *italics* first, before we go on to discuss *if* and *how* this requirement is covered by the COMET architecture.

5.1 Coverage of Global Requirements

- ***"The content must be treated as a primitive itself.*** *The architecture must be oriented to deal with all aspects of content natively, facilitating the access and distribution of contents. Support for safe, based on trusted content publication, friendly and fast content retrieval for consumers through the COMET architecture and mediation functionalities is required."*
  - The CMF offers an interface to the client to request for content using *Content Names* or *Content IDs*. The CRF also offers a unified interface to the Content Publishers for the publication process. Based on the *content characteristics* (e.g., the *Content Properties* as discussed previously in the decoupled approach), the underlying paths are configured accordingly. Therefore, the whole path is built based on the content's requirements, which makes guarantees that content is *treated as a primitive* indeed. The CRF ensures that Content publication is done through trusted third party entities, which further guarantees safe and trusted transactions with the COMET system.
- ***"A global content naming and addressing scheme should be supported by an infrastructure capable of scalable content search and resolution.*** *The global content-aware mechanisms must be able to handle efficiently large amounts of content, being able to support significantly more objects than those handled by today's largest Content Distributors (YouTube, Flickr, Apple Store, for example). The protocols to be developed by the project will be capable of scaling to the order of billions (10^9) of content objects."*
  - Our Content Naming Scheme is discussed in detail in D3.1 [2]. There, we show that because of the fact that objects are identified by *Content Names* or *Content IDs*, the corresponding header entries can be arbitrarily big in order to accommodate billions of published content. The scalability of the resolution process within COMET has not been proven yet at this early stage of the project; our views as to why the resolution functions are indeed going to scale are discussed in D3.1 [2], while scalability proofs are going to be provided in the next year of the project.
- ***"The COMET system should be open for future evolution of the Internet.*** *This can be achieved by the modularity in the design of different components and with a flexible high-level architecture."*
  - None of the functions, functional blocks or entities included in the COMET architecture prevents extension or enhancement in order to accommodate future evolution of the Internet. The design of all functions and components are generic and flexible enough to be easily extended in order to accommodate more functionalities.
- ***"Support for gradual and economical embracement of the COMET system by ISPs.*** *The designed architecture for content mediation and the associated mechanisms for content discovery, resolution and access must be scalable to be deployed in the largest*

ISPs, consisting of the order of hundreds of point of presence (PoPs) and core routers. These mechanisms and protocols should be applicable for content distribution at Internet-scale, involving autonomous networks of the order of tens of thousands of ASs."

- As discussed earlier, the functions introduced earlier and the grouping to entities will allow the corresponding planning from the ISPs, as for the placement and the number of these entities per AS. Having said that, the architecture is envisioned to scale well, the same way as the current AS-/BGP-based Internet scales today and has been designed taking into account this requirement. Scalability has not been proved yet at this point, but will be proved in the near future.

- *"The content-aware mechanisms designed and developed for the network, when orchestrated by novel Content Mediation Plane (CMP) algorithms and protocols, **should facilitate the involvement of, potentially, all Internet users as Content Creators**. Thereby, creating the opportunity of a new, all-encompassing market where millions of small, medium and large Content Providers have access to efficient content distribution capabilities to reach billions of potential Content Consumers, taking advantage of a reduction of required resources, mainly bandwidth and processing capacity."*
  - The COMET Architecture supports User-Generated Content. That is, all Internet users, be it simple users, or big content providers can publish content into the system by the unified interface that the CRF provides to the Content Publishers. The Content naming scheme has been designed to allow the presence of potentially, all Internet users as Content Publishers. The details are given in D3.1 [2].

- *"The COMET system will support handover mechanisms which allow **a graceful switching of the content delivery path without impact on the application-layer**."*
  - This feature is not yet supported by the COMET architecture. Investigations will follow during the second year of the project, in order to decide whether such mechanisms can be accommodated in the COMET system.

5.2 Coverage of Requirements for the Content Consumers (and Content Clients)

- ***"Access to the contents must be independent from the content location.*** *The naming architecture should guarantee location-independence, which in turn would guarantee smooth transition from today's host-centric to a future content-centric Internet."*
  - Content in COMET is accessed using *Content Names* or *Content IDs*. The structure of both of these identifiers is not based on the content location. The actual location of the content is resolved within the COMET system and in particular, in the CRF, which makes content resolution *location-independent*.

- ***"The content identifier must be the same for different ways of distribution and nature of the content.*** *Also, different copies of content will be identified by the same Content-ID. It is, however, responsibility of the Content Providers to explicitly register the new copy of the content as such."*
  - All copies of the same content are indeed identified by one Content name or Content ID. The specifics of these operations are given in D3.1 [2]. All the parameters associated to a content, e.g., QoS requirements, ways of transmission etc., are given by the content publisher at the time of the registration of the new content.

- *"**The Content Consumers must access the content in the same way as in current Internet** i.e. achieving user unawareness."*
    - Both approaches in the COMET architecture allow "click-to-consume" user interfaces to access the content in the same way as in current Internet with hyperlinks. In the coupled approach, the "click-to-consume" will be translated directly into a content request which will reach, through the hop-by-hop content resolution, the Content Server. The decoupled approach follows a two round-trip approach, with two requests from the Content Client, the first one to get the content properties and the second one to perform the application-level request. Nevertheless, this whole process of two requests is transparent for the end user and perceived as a one-step procedure.
- *"The Content Client could optionally declare its capabilities during content resolution phase, but **it is up to the COMET system to decide how to deliver the content to the Content Consumer**."*
    - The decision on how to deliver the content is the job of the CMF. For the *Decoupled Approach*, the CMF, based on the information that it gathers from the CRF (content properties), PMF (offline, long-term path condition) and SNMF (online, near real-time network and server condition), it makes the decision of how to deliver the content to the client. The path(s) is then configured accordingly. For the *Coupled Approach*, the delivery path is built while resolving the content request and thus, based on the business relationships between the intermediate ASes (possibly gathered from PMF). The path optimization function in the *Coupled Approach* may use the information from SNMF.
- ***"The Content Client will obtain all the parameters necessary to invoke the application level requests."***
    - In the decoupled approach, there is no need for modification of current applications in order to fit into the COMET architecture since the Content Client will receive the content properties after the content resolution sub-operation. In the coupled approach, the two round-trips are combined into one, thus forcing some kind of adaptation in the application or transport protocols to provide to the Content Server enough flexibility to deliver the content.

5.3 Coverage of Requirements for the Content Providers (and Content Servers)

- *"**There must be an interface that allows the Content Providers to update the content properties** (content location, server load, way of distribution, etc.)"*
    - This is accomplished by the unified interface between the Content Publisher and the CRF. This interface allows for "UPDATE" messages regarding the content properties. More details about the unified interface and the update messages are provided in D3.1 [2]. The server load information is provided by the SNMF function as described earlier.
- *"**The Content Provider should be able to establish policies** to enforce the way to publish and deliver the contents to the Content Consumers."*
    - The CRF allows Content Publishers to express their preferences related to publication and delivery of their contents. After that, the CMF, through the *Path Configuration* process, is able to configure the path that will be used to deliver these contents from the Content Servers to the Content Clients.

5.4 Coverage of Requirements for the CMP (mediation layer requirements)

- *"**There must exist a global content resolution architecture** for efficient and scalable name and content resolution."*
    - This is fully covered by the unified content access intelligence added to the COMET system by the two content resolution approaches. In particular, the CMF with the help of CRF can guarantee global content resolution. The details are given in D3.1 [2], while the scalability has not been proven yet.
- *"**There should be an integrated traffic and resource management solution compatible with the content resolution architecture** to increase network efficiency and content delivery in order to reduce network congestion on the most highly loaded links."*
    - This requirement is covered by the cooperation of a couple of functions: PMF gives to the CMF offline info about the availability of underlying paths. The SNMF, on the other hand, deals with near-real time, online information. Finally, the CMF makes the decision taking into account the increase of network efficiency and content delivery and configures the corresponding delivery paths.
- *"**There should be an information gathering system in the CMP** for collection of various performance metrics on networks and servers. This is going to be implemented in the COMET Monitoring Module."*
    - The monitoring module that gathers online, near real-time information is realized in the SNMF, which gathers network and server monitoring information and feeds it to the CMF.
- *"**The protocol interfaces between the CMP and the Content Providers, publishers and end user devices must be efficient.** The user terminals should be able to send their content consumption requests through these interfaces, and the Content Providers must announce their server condition and the information about the contents they publish using these interfaces. To complete this requirement, some others have been extracted from the use cases:"*
    - The collaboration between the different functional blocks of the architecture will guarantee efficient content publication, resolution and delivery. As discussed, the different functional blocks, provide information to the CMF, which taking into consideration the client's request requirements and the provider's content characteristics, makes decisions on the most appropriate path to follow.
    - ***"The CMP must be able to dynamically modify the information related to the location of the servers in the content record."***
        - Such information, once provided by the Content Publisher, will be dealt with by either making use of specific "UPDATE" messages sent to the CRF, or by new "publish" commands.
    - ***"The COMET system must offer to the Content Provider the possibility of registering different ways of distribution."***
        - This information has to be provided by the Content Publisher and will be included in the content properties of the content in question.
- *"**The CMP in an ISP must be aware of network conditions** in order to take decisions oriented to reduce the latency in content retrieval that is due to network failures, network congestion or server load."*
    - There are two functionalities that gather network information and feed it into the CMF in order for the latter to choose the right delivery paths accordingly. These are,

the PMF, which gathers offline, long-term information and the SNMF, which gathers near real-time information. Network failures, network congestion and server load information relate mainly to near real-time conditions that are provided by the SNMF block.

- ***"There should be some kind of interaction between the Content Mediation Plane and the Content Forwarding plane to enforce content delivery."***
    - The SNMF and the PMF provide the required information to the CMF, which in turn enforces the underlying paths accordingly and prepare the content delivery. The functionality of the Content Forwarding Plane that receives these rules and enforce them in the network to prepare the content delivery is the CAFF.
- *"**The CMP, upon the content request from a user device, should be able to request capabilities to enhance or facilitate the QoS and multicast in the network** for the delivery of that content to that user device."*
    - As stated previously, the Content Mediation Plane through the CMF function takes decisions about the path and prepares it for the content delivery. On the Content Forwarding Plane side, the CAFF, after gathering the required info from the CMF, regulates traffic according to the rules.

5.5 Coverage of Requirements of the CFP (network layer requirements)

- ***"There must be a content forwarding architecture able to perform content-based forwarding at speeds similar to the ones in IP-based forwarding."***
    - The CAFF implemented in CAFEs interact with normal IP-routers in order to regulate traffic accordingly and deliver the content to the content client and it is supposed to work at speeds similar to today's IP-based paradigm.
- *"The elements in the CFP should support **QoS-aware content delivery**."*
    - The CMF is able to enforce QoS-related rules in the CAFEs. These latter entities, by the CAFF functionality they implement, are able to enhance the content delivery. Details on the operations carried out therein are given in D4.1 [3].
- *"The elements in the CFP should support **point-to-multipoint content delivery**."*
    - Point-to-multipoint content delivery is natively supported by the *Coupled Approach* described previously and is detailed in D3.1 [2] and D4.1 [3]. For the *Decoupled Approach*, the required intelligence is going to be added to the CFP to support such kinds of content distribution. The specifics of how this is going to be accomplished are to be included in later documents, while preliminary views are included in D4.1 [3].
- *"**Content may be cached in the network** to optimise network resource usage."*
    - COMET-specific caching approaches have not been implemented yet in CAFEs, but an initial study has been performed in order to choose the best one available. Details of this study are included in D4.1 [3].
- ***"There should be an interaction between the CFP and the CMP to provide information on network conditions and, optionally, routing information."***
    - The monitoring SNMF module provides network condition information to the CMF. In addition, information about routing is provided by the PMF in an offline, long-term fashion.

5.6 Summary table

The following table gathers the fulfillment of the system requirements that have been explained in the previous sections. As can be seen more than 70% of the system requirements defined in D2.1 [1] are already covered by the COMET Architecture presented earlier in Section 3 of the present document.

ID	Category	System requirement	Coverage
1	Global	Content as a primitive	Already covered
2	Global	Global content naming and addressing	Already covered
3	Global	Open for future evolution of the Internet	Already covered
4	Global	Scalable to be deployed in the largest ISPs	To be covered
5	Global	Involvement of all Internet users as Content Creators	Already covered
6	Global	Graceful switching of the content delivery path without impact on the application-layer	To be covered
7	Content Consumer	Access independent from content location	Already covered
8	Content Consumer	Content ID independent from way distribution and nature of content	Already covered
9	Content Consumer	User unawareness	Already covered
10	Content Consumer	Content Client able to declare his capabilities	Already covered
11	Content Consumer	Content Client will obtain all necessary parameters to invoke the application level requests	Covered in the decoupled approach
12	Content Provider	Interface to update the content properties	To be covered
13	Content Provider	Capability of establishing policies to enforce the way to deliver contents	Already covered
14	CMP	Global content resolution architecture	Already covered
15	CMP	Integrated traffic and resource management solution to increase network efficiency and content delivery	Already covered
16	CMP	Information gathering system	Already covered
17	CMP	Efficient protocol interfaces	To be covered
18	CMP	Capability of dynamically modify servers location information	Already covered
19	CMP	Possibility of registering different ways of distribution	Already covered
20	CMP	Network conditions and routing information awareness	Already covered
21	CMP	Interaction between the Content Mediation	Already covered

		Servers and the Content Aware Forwarders to enforce content delivery	
22	CMP	CMP able to request the enforcement of QoS and multicast in the network	Already covered
23	CFP	Content forwarding architecture able to reach IP-based forwarding speeds	To be covered
24	CFP	Elements in CFP able to support QoS-aware content delivery	Already covered
25	CFP	Elements in CFP able to support point-to-multipoint content delivery	To be covered
26	CFP	Content may be cached to optimize network resource usage	To be covered
27	CFP	Interaction between the CFP and the CMP to provide information on network conditions and, optionally, routing information	Already covered

Table 1: System Requirements fulfilment in the COMET architecture

6 Conclusions

We have presented the *High-Level Architecture of the COMET System*. We based our design on the *analysis and synthesis* methodology and split the basic COMET *operations*, into *sub-operations* and *processes* following a top-down approach. Then, we grouped these *processes* into *functional blocks* and these *functional blocks* into *entities*. The main *functional blocks* of the *Content Mediation Plane* (CMP) achieve *content resolution*, through the *Content Resolution Functional block (CRF)*, *Path Management*, through the *Path Management Functional block (PMF)*, *Server and Network Monitoring*, through the *Server and Network Monitoring Functional block (SNMF)* and finally, *Content Mediation*, through the *Content Mediation Functional block (CMF)*. In the *Content Forwarding Plane (CFP),* there is the *Content-Aware Forwarding Functional block (CAFF)*.

These *functional blocks* are grouped into one or more entities, depending on the mediation approach followed. That is, we are currently investigating two approaches: the decoupled approach and the coupled approach, as described extensively in Section 3 of the present document. Further research is required in order to assess the exact scalability and performance properties of these two approaches.

Based on the overall COMET architecture, the borderline between WP3 and WP4 within the project is now clearer. We have elaborated on these issues in Section 3.5, where we described the differences both in terms of the COMET basic operations and in terms of the *functional blocks*.

The COMET Use Cases have been mapped onto the overall Architecture as this evolved through the course of the project. We found that most use cases can be accommodated under the COMET Architecture.

Finally, we discussed how the System Requirements, defined in D2.1 [1], can be covered by the *High-Level COMET Architecture*.

Further research needs to be done in order to assess the scalability properties of the proposed architectures, for instance. The specifics of the mechanisms and algorithms for *Content Publication, Content Resolution and Content Delivery* are included in subsequent Deliverables D3.1 [2] and D4.1 [3].

7 References

[1] First COMET Deliverable, *"D2.1: Business Models and System Requirements"*, The COMET Consortium, July 30th 2010.

[2] COMET Deliverable, *"D3.1: Interim Specification of Mechanisms, Protocols and Algorithms for the COMET Mediation System "*, The COMET Consortium, November 30th 2010.

[3] COMET Deliverable, *"D4.1: Interim Specification of Mechanisms, Protocols and Algorithms for Enhanced Network Platforms"*, The COMET Consortium, November 30th 2010.

8 Abbreviations

CAFF	Content-Aware Forwarding Function
CAFE	Content-Aware Forwarding Entity
CDN	Content Distribution Network
CFP	Content Forwarding Plane
CME	Content Mediation Entity
CMF	Content Mediation Function
CMP	Content Mediation Plane
CMS	Content Mediation Server
COMET	COntent Mediator architecture for content-aware nETworks
CNAME	Canonical Name
CRE	Content Resolution Entity
CRF	Content Resolution Function
CRME	Content Resolution and Mediation Entity
DNS	Domain Name System
IP	Internet Protocol
ISP	Internet Service Provider
NAT	Network Address Translation
P2P	Peer to peer
PMF	Path Management Function
QoS	Quality of Service
RAE	Routing Awareness Entity
SNMF	Server and Network Management Function
URL	Uniform Resource Locator
VoD	Video on Demand

9 Acknowledgements

This deliverable was made possible due to the large and open help of the WP2 team of the COMET project within this STREP, which includes besides the deliverable authors as indicated in the document control. Many thanks to all of them.