COntent Mediator architecture for content-aware nETworks

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

Deliverable D6.2
Prototype Experimentation and Demonstration

The COMET Consortium

Telefónica Investigación y Desarrollo, TID, Spain
University College London, UCL, United Kingdom
University of Surrey, UniS, United Kingdom
PT PLC, PT, 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:

Mr Michael Georgiades
PrimeTel, michaelg@prime-tel.com
The Maritime Centre
141 Omonia Avenue
3506 Limassol

Phone: +357 25 867000
E-mail: comet@comet-project.org

Document Control

Title: Demonstration Scenarios and Test Plan

Type: Public

Editor(s): Michael Georgiades

E-mail: michaelg@prime-tel.com

Author(s): Michael Georgiades (PTL), Lenos Andreou (PTL), Eugeny Grep (PTL), David Flórez (TID), Nacho Conde (TID), Piotr Wiśniewski (WUT), Andrzej Bęben (WUT), George Petropoulos (ICOM); Wei Chai (UCL)

Doc ID: 20022013-D62-v1.0.docx

AMENDMENT HISTORY

Version	Date	Author	Description/Comments
v0.1	September 11th	Michael Georgiades	First Version of ToC
v0.6	November 14th	David Flórez	Initial contribution to federated testbed description, both global and TID
v0.7	November 16th	David Flórez	Description of Use Cases
v.08	November 23rd	David Flórez and Nacho Conde	Description of VPN servers and quaga configuration at TID. Description of the ENVISION-COMET joint scenario
v0.9	November 26th	Piotr Wiśniewski and Andrzej Beben	Description of WUT testbed. Description of CSR tests.
v.91	November 30th	Michael Georgiades	Description of PTL Testbed, PTL, Appendix, Editorial
v.92	December 10th	Piotr Wiśniewski and David Florez	Sections 3.5.1, 3.5.1.3, 3.5.2, 3.5.2.3, 3.5.3.3, 3.5.4.3, 3.5.5.3, 3.5.6.3, 4.1.2, 4.2.3, A.2.8, A.2.9- A.2.10.9
v.93	December 13th	George Petropoulos, Lenos Andreou	Section 5.3, Updates to Section 3 and Annex A.
v.94	December 18th	Piotr Wiśniewski and Andrzej Beben Michael Georgiades	Input on Performance Evaluation Section Executive Summary, Introduction, Conclusions, Input to Performance Testing
v.95	December 19th	All	Section Review
v0.96	December 20th	Michael Georgiades	Added Section 4.3
v0.97	January 30th	All	More Validation Testing Updates
v0.98	February 15th	Michael Georgiades and Lenos Andreou	Additional Performance Tests in Section 4.2
V0.99	February 18th	David Flórez	Review
V1.0	February 20th	David Flórez	Final Version

Table of Contents

(This page is left blank intentionally.)

1 Executive Summary

This deliverable summarises the final description of the complete COMET system that has been deployed and tested based on the predefined test cases discussed in D6.1 228[7]. For ease of reference to D6.1 [7] a similar content structure has been maintained. Hence the deliverable's first two main sections describe all the involved configuration and setup together with the associated functional and performance testing for the overall evaluation of the framework. The two sections are as follows:

- **Configuration and setup of the final COMET Federated Testbed**, describing how the decoupled approach of COMET was realised and the Federated Testbed setup between TID, PTL and WUT, the associated Local Testbed configurations, administrative domain setup at each site and the necessary interworking setup configurations. A description of configurations regarding the use cases is also included in this section as it involves each of the sites, hence the complete Federated Testbed and associated prerequisites on each site. For each site we describe the local testbed used in turn describing the topological layout, the forwarding key configurations used, subnets created, path configurations, class of services for the content clients and servers, the selected content names and content source used, the decision algorithm tunings and the necessary tunnel configurations between the sites.
- **Functional and Performance Testing of the COMET Federated Testbed**, which is split into two subsections, namely, Functional Tests and Performance Tests:
  - **Functional Tests.** These are based on the functional tests described in D6.1 [7] which were derived from the selected use cases Use Case 1 (UC1) related to adaptable and efficient content distribution and Use Case 4 (UC4) related to P2P Offloading. For UC1 separate tests where undertaken for each use case covering routing awareness, content publication, content delivery, name resolution, obtaining server load, selection decision process, path configuration and content consumption. Moreover we examine situations on what happens in the case of adding a new server, what happens in the case a content owner changes the type of CSs distribution of the content, when content is accessed with same name from different domains, the class of service rules management by the CME, obtaining optimal path selection, selection based on server load, and finally two tests to evaluate the operation of point to multipoint streaming. For UC4 different functional tests were undertaken including RAE information gathering, content record publishing with content sources describing the characteristics of the streaming servers, streaming distribution and CS selection and what happens in the case of high load in streaming servers.
  - **Performance Tests**. These are based on the performance tests discussed in D6.1 [7] aiming to cover a performance evaluation and stress test the different components of the Federated testbed namely (SNME, RAE, CAFE, CRE and CME). Note that some of these were tested locally within a particular local testbed and some across the Federated Testbed on a case by case basis. Along with evaluation of the implemented framework, performance measurements obtained during the evaluation were also utilised as benchmarks in support and as input to the simulation studies on large-scale scenarios undertaken in WP5.
  - **Validation and Verification**. This relates to checking in that the D2.1 [1] quantitative requirements as well as and performance metrics (qualitative requirements) discussed in D6.1 [7] have been met.

The final section of the deliverable describes the outcome of the collaboration with the ENVISION [16] project:

- **COMET-ENVISION integration:** Here we describe the integration of the ENVISIONs CINA component into the COMET's federated testbed for providing addition cost parameters in support of path selection. It begins with a summary of the COMET-

ENVISION joint scenario and how cost information from the CINA system can refine the path decision algorithm implemented in the COMET CMEs. This is followed by a supportive layout on how it was integrated in the federated testbed, a description of the use-case used for evaluation, extension requirements on CME for the CINA interworking to be possible and finally a description of the joint testbed's functional test evaluation.

Appendix A provides thorough details of the Federated Testbed's Raw Data and the addressing schema, forwarding keys, IP prefixes, Mappings, Path Configuration settings, Content Names selection, open VPN and Quagga Configuration for each of the testbeds together with the use case configuration that could be used as settings reference for future related experimentation.

Based on the aforementioned, Deliverable D6.2 gives a thorough description of all the configurations required to setup the federated testbed to evaluate the decoupled approach as well as an extensive functional and performance test evaluation which illustrated that the COMET system is functional, operational and can scale. Performance benchmarks used here have been exploited also for analytical and simulation studies in WP5 as reflected in D5.2 [6]. Finally the COMET-ENVISION interworking solution has been shown to provide a more informative path selection with the addition of further cost criteria.

2 Introduction

D6.2 is the second and final deliverable of WP6 which has tested and demonstrated the COMET system based on the design, implementation and integration achieved in the previous WPs together with the use cases and associated functional and performance tests as defined in D6.1 [7]. What is referred to as Federated Testbed aims to deploy the COMET's decoupled approach in an as close to a real system as possible. To achieve this, the system setup emulated several ISPs as in the real Internet. The topology is set up in such a way to also allow for a content client to request content from a content server at a remote ISPs while involving at least one intermediate ISP. Moreover multiple network paths and multiple content servers are set up to allow for all necessary selection options in terms of class and quality of service.

Section 3 describes the final layout of the COMET federated testbed, comprising three interconnected partner local testbeds (TID, WUT and PTL) focusing mainly on the necessary configurations to achieve this. A similar structure is maintained as in D6.1 [7] for ease of referencing, hence for each local testbed there is an initial subsection on the overview of the Federated Testbed layout, a detailed configuration of a testbed's setup including forwarding key configuration, a description of the necessary interconnection configurations including peering CAFE key mappings, class of service (CoS) mappings, content names and sources used and finally tuning of the decision algorithm. Finally there are two common subsections for the overall federated testbed on remote access configuration and a summary of the selected use cases for the functional and performance tests to follow.

Section 4 describes the extensive testing achieved in WP6 based on the integrated federated testbed with inputs from WP3, WP4, WP5), the test cases and functional tests defined in D6.1 [7] for the uses cases defined in WP2 and the performance metrics defined in T6.2. The first part of the functional tests focuses on Use Case 1 related to adaptable and efficient content distribution with functional tests on routing awareness, content publication, content delivery, name resolution, server load, path decision process, path configuration, adding a new server, changes to the CSs distribution of the content by the content owner, content consumption, accessing content with same name from different domains, management of class of service rules by the CME, optimal path selection, selection based on server load, and finally two functional tests to evaluate the operation of point to multipoint streaming. The second part on the functional testing focuses on Use Case 4 hence the P2P offloading case scenario with functional tests on routing awareness related to RAE, another on distributed content publishing by a content owner, streaming distribution and finally how the system operates when the load is high in streaming servers.

The second part of Section 4 focuses on Performance Testing. For this work the COMET team has executed an extensive performance evaluation study of the COMET system. Moreover we have created a number of stress tests to stress the limitations of the system and allow checking its scalability. These included stress tests for all main components, namely, Content Resolution Entitiym (CRE), Content Mediation Entitity (CME), Server and Network Monitoring Entity (SNME), Route Awareness Entity (RAE), Content-Aware Forwarding Entity (CAFE), Content Server (CS) and Content Client (CC). The majority of the stress performance related tests were able to be tested locally on a particular testbed but in some cases the global testbed was used.

Section 5 focuses on validation and verification in that the D2.1 [1] quantitative requirements as well as and performance metrics (qualitative requirements) discussed in D6.1 [7] are met.

Section 6 of the deliverable discusses the joint use case of enabling COMET system to interact with the ENVISION system for a more informative decision making when it comes to path selection. This section summarises the work undertaken, a layout of the integrated testbed, the use case to test the scenario, the required changes in the COMET CME to support ENVISION CINA parameter inputs, and finally functional tests to evaluate it.

The deliverable closes with several major conclusive remarks in Section 6 and some words on the outcomes of this work and on future research.

There is also a four part Appendix which is referenced throughout the document containing more specific details on each of the testbed's setup and more complete configuration settings established as well as specific configurations related to the two use cases addressed in the deliverable.

3 Final COMET Federated Testbed

This section describes the final layout of the COMET federated testbed, comprising three interconnected partner testbeds (TID, WUT and PTL). Over the COMET federated testbed a selected set of use cases from D2.1 [1] will be demonstrated, following the guidelines already sketched in D6.1 [7], in order to prove the validity of the COMET approach. The federated testbed will also be the basic frame for functional testing and part of the performance characterisation (see section 4.2 in this same deliverable for more details and test results)

This section is structured in the following subsections:

- Overview of the Federated Testbed layout.
- Detailed configuration of each one of the partner testbeds (WUT, TID, PTL).
- Description of interconnection procedures and configuration required in each partner testbed.
- Description and configuration of the remote access.
- Brief explanation of how the use cases selected in D6.1 [7] are intended to be demonstrated on the federated testbed D6.1 [7], based on the use of the federated testbed remote access.

Note that the federated testbed is allocated for the demonstration of the decoupled approach only, not the coupled one.

3.1 Federated Testbed Layout and Concepts

The next figure shows the final layout of the federated testbed, as it has been set up to demonstrate the choice of use cases described in D6.1 [7].

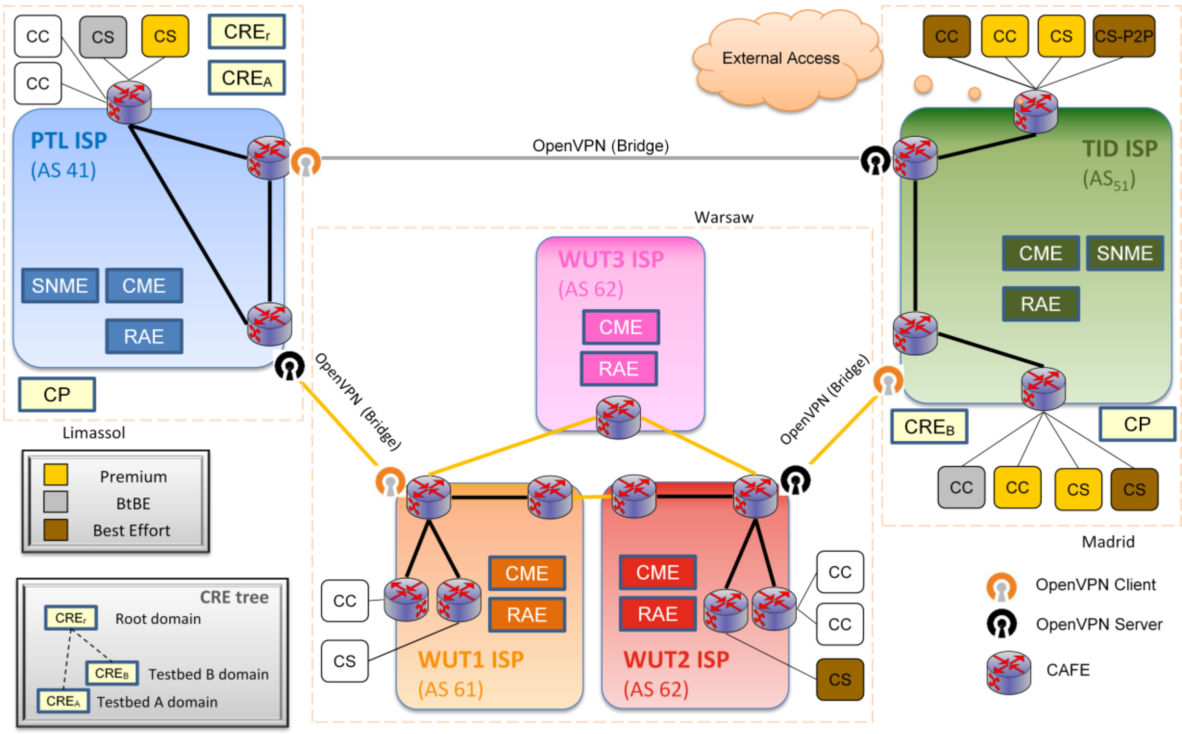


Figure 1: Final Layout of the Federated Testbed

Next two subsections will briefly explain the main concepts behind the COMET federared testbec layout and the main action taken in its configuration

3.1.1 Federated Testbed Concepts

The federated testbed consists of three testbeds located at partner premises, Madrid/TID, Limassol/PTL and Warsaw/WUT. It has been configured to work either in pure IPv6 or IPv4 mode, but not as a mixed environment. All entity configurations are separately described both for IPv6 and IPv4.

Local testbeds are connected by OpenVPN [17] IPv4 tunnels in bridge mode, which are configured to encapsulate both IPv4 traffic and IPv6 traffic. Each OpenVPN bridge involves an OpenVPN Server and an OpenVPN Client, which have been configured to create a wheel, where a local testbed launching a client to connect to another local testbed, deploys a server for accepting connections from the third one. In summary:

- Tunnel TID-PTL. TID as server, PTL as client.
- Tunnel PTL-WUT. PTL as server, WUT as client.
- Tunnel WUT-TID. WUT as server, TID as a client.

Each testbed stands for an Internet Service Provider (ISP), except in the case of WUT, which hosts three of them to allow for more complex routing possibilities. Therefore:

- PTL has a single associated Autonomous System (AS), named AS 41.
- TID also has a single associated AS named AS 51.
- WUT incorporates three of them, named respectively AS 61, AS 62, and AS 63.

A second OpenVPN server has been setup in TID testbed in order to allow external access to the federated testbed, thus enabling the capability of demonstrating COMET outside partners' premises. This second OpenVPN server is deployed in one of the edge CAFEs located at TID. The function of an edge CAFE is explained below, but here it suffices to say that remote CCs, can be dynamically attached to that edge CAFE, displaying a behaviour identical to any CC physically attached to that CAFE.

In terms of demo purposes, the external access also defines a polarisation of the federated testbed, with contents being transmitted to CCs at TID from CSs located in the other two testbeds. This polarisation does not apply to functional/performance tests, though.

A central concept of COMET is the Class of Service (CoS), which helps to segmentate customers in different categories according to their subscribed SLA. Three CoSs have been defined in COMET:

- Pr (Premium).
- BtBE (Better than Best Effort)
- BE (Best Effort)

The CME decision algorithm has been configured so that VPN links between partner testbeds preferently transmit traffic belonging to one of the CoS, diverting lower CoS to other paths when necessary. In short, for demo purposes and from the point of view of TID CME:

- The link TID-PTL is primarily allocated for BtBE traffic (silver).
- Links traversing WUT are allocated to Pr traffic (Gold).
- BE traffic can cross through any path, as long as it does not impair any other priority traffic.

In terms of decoupled entities, CREs are not associated with a specific ISP (hence the term decoupled). A hierarchy is defined among CREs, similar to DNS, from a root CRE to authoritative CREs which store Content Records located in one or several domains. In the federated testbed, the CRE root is located at PTL testbed, and two authoritative ones are deployed at TID and PTL, respectively, while no CRE is deployed at WUT premises.

Each ISP hosts a RAE. The ISP RAE informs neighboring RAEs of the known paths leading to that ISP, qualified with Quality of Service (QoS) parameters. After a transitory period, each RAE will be aware of how to reach any other ISP.

Each ISP hosts a CME. The CME resolves a Content Name into the address of a CS distributing the content, using the resolution capabilities of the CRE, as well as selecting and configuring an optimal path from the selected CS to the requesting CC.

CAFEs are intelligent routers able to intercept traffic at its source (the CS), route it through the path selected by the CME and ensure QoS for the content characteristics. This interception will be only by applied to traffic labeled as Pr or BtBE CoS. CAFEs can play two different roles:

- Edge, serving CSs and CCs in the access networks
- Border, managing traffic in the links connecting neighboring ISPs.

The distribution of CAFEs in the federated testbed is illustrated in the

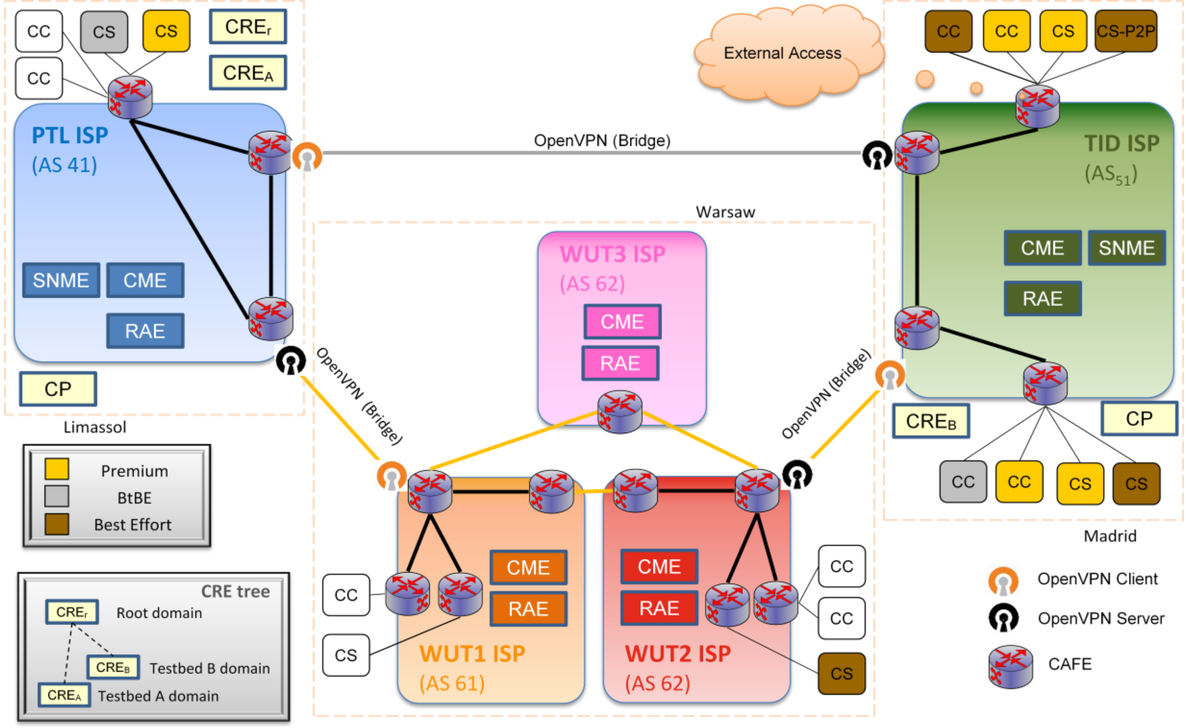


Figure 1 and can be summed up as follows:

- PTL has two border CAFEs (to TID and to WUT) and one edge.
- TID has two border CAFEs (one linking to PTL and other one to WUT) and two edge ones, in order to serve two different set of CCs and CSs. Remote CCs accessing via the OpenVPN remote access will be attached to one of the edge CAFEs, and behave identically to those physically attached to that CAFE.
- WUT has two "external" border CAFEs for managing traffic to/from PTL and TID, three "internal" border CAFEs for managing traffic among its internal ISPs. Besides, two edge CAFEs, one for serving CSs and other for CCs, deployed at both AS 61 and AS 62.

A number of CSs serving contents mediated by COMET have been deployed throughout the federated testbed. They can be classified according two criteria: Distribution means and CoS:

- Classified by distribution means, the CSs can be:
  - Streaming/Video on Demand (VoD). Streaming is implemented by using VLC [14] with HTTP streaming both for IPv6 and IPv4. VoD is achieved by means of Apache Web Servers. Typically this sort of server runs Linux OS.

    - P2P. This service is implemented by using µTorrent [15]. In this case, the CS mandatorily runs windows OS.
- Classified by CoS, there are three types of servers:
    - Pr CSs.
    - BtBE CSs.
    - BE Cs.

The minimum CSs distribution in the federated testbed to cover the requirements of the envisaged demo cases (reasons will be explained in section 3.7) is as follows (although more CSs can be added as required in any location):

- In PTL, there are one Pr Streaming/VoD CS and one BtBE Streaming/VoD.
- In TID, there are two Pr Streaming/VoD CSs, one BE Streaming/VoD and, one BE P2P.
- In WUT/AS62, there is one BE CS.

A number of CCs have been deployed in each ISP to test retrieval of contents mediated by COMET. These CCs are assigned to different CoS in order to cover all different possibilities. The minimum distribution for the envisioned demo, taking into account the testbed polarisation, is as follows (although more clients can be added as required):

- In TID there are two Pr CCs, one BtBE CC and one BE CC
- At least one CC from each CoS can be connected via the remote access to the federated testbed for demo purposes

A SNME is required in each ISP were CSs are deployed, so that local CME is informed about server statuses in the ISP it manages. SNMEs have been consequently deployed wherever CSs are in operation, namely PTL, TID, and WUT (AS 61).

Any ISP containing an authoritative CRE also includes a CP in order to allow the publication of Content Records in these elements. CPs are thus deployed at PTL and TID.

One of WUT's domains includes the CSR entity. The CSR supports local multicast in the client's domain. It receives content streams from border CAFEs and sends unicast or multicast streams to a number of CCs.

3.1.2 Required Configuration Actions

Once explained the federated testbed layout and in order to understand the configuration information presented in the following sections, it is important to keep in mind that at least the following configuration actions must be carried out each testbed, in order to achieve that a CC can retrieve contents from a CSs located either inside its own ISP or remote ones.

In each CAFE, either border or edge, a set of keys need to be defined. A key assigns a hexadecimal code (i.e 0xff) to a pair of Ethernet interfaces in the network link between two directly connected CAFEs: the outgoing Ethernet interface in the CAFE where the key is created and the incoming Ethernet Interface of the CAFE at the other link end. Keys then define a traffic direction through the link.

The RAE has to be configured with the QoS parameters of paths, either inside each local tested, either linking to neighbouring ISP. Three different configuration activities have to be carried out:

- First, for each QoS parameter (Packet Loss, Packet delay, and Bandwidth) and each CoS (Pr and BtBE), a set of targets have to be defined in order to guide the CME in the decision process. The targets are as follows:

    - Reservation Level or a measure of how much of this parameter can be booked for a single flow
    - Aspiration Level or importance of the parameter in the decision algorithm (the lower the value the most important)
- Secondly, the internal paths have to be characterised. This means that for each path connecting subnets where CC & CCs are deployed and for each CoS the following parameters have to be specified
    - Bandwidth of the path.
    - Packet Loss.
    - Packet Delay.
    - Source Subnet/Sink Subnet.
- Finally, the same characterisation has to be carried out for each path to neighbouring domains. The information that has to be provided for each path is:
    - Bandwidth of the path.
    - Packet Loss.
    - Packet Delay.
    - Source AS/Sink AS.

CME requires a more complex configuration, which can be split in five different tasks:

- First, transit paths need to be configured in the CME. A transit path is any path inside a testbed linking any two given CAFEs. In order to avoid an unnecessary explosion of paths, only those end to end paths linking two edge CAFEs and an edge CAFEs with a border CAFE need to be defined.
    - Given two CAFEs, a path has to be defined for each traffic direction.
    - For each traffic direction, as many paths as CoS have to be created (only Pr and BtBE).
    - Each individual path needs these parameters to be completely configured:
        - The list of CAFEs keys the content traverse from CAFE source to CAFE sink. The list is composed in reverse order to the traffic direction.
        - The CoS of the path.
        - The maximum capacity of the link, end to end.
- Secondly, peering paths have also to be defined. A peering path is one that links a border CAFE in the ISP with a border CAFE in a neighbouring CAFE. For each link, a path has to be specified for each traffic direction and there will also be as many paths per direction as CoSs (only PR and BtBE). The parameters to be specified for each path are the following:
    - AS joined by the link.
    - CoS of the path.
    - Source and sink CAFE.
    - Capacity of the link.
    - CME serving the neighbouring ISP.
- The subnets attached to each edge CAFE also have to be defined, so that CME can find out which edge CAFE serves which CCs/CSs
- Each CC has to be assigned a CoS in order to enable its CME to assign the right CSs/Path for queried contents.
- Finally, the CME decision algorithm parameters have to be defined. Basically there are six parameters that can be modified by assigning different reservation/aspiration levels:
    - Number of candidates or how many paths/servers will be taken into account by the decision algorithm.

- Server Load.
- Path Length. Maximum hops tolerated in a path.
- IPTD. Maximum Packet Delay.
- IPLR. Packet Loss Ratio.
- BW. Bandwidth.

The next sections describe how each of the partners' testbeds have been configured following these guidelines.

3.2 *TID Local Testbed*

This section describes the layout of TID local testbed. Note that some references to federated testbed interconnection have been included to group similar pieces of information, even though they will be described in detail in section 3.5.

3.2.1 Local Testbed Layout

The following figure depicts the layout of TID local testbed. No reference to IP, either IPv6 or IPv4, or MAC addresses have been made in the figure, whose elements will be referred by name. Since the Ethernet interfaces are required in the CAFE forwarding key generation they have been labeled in the figure:

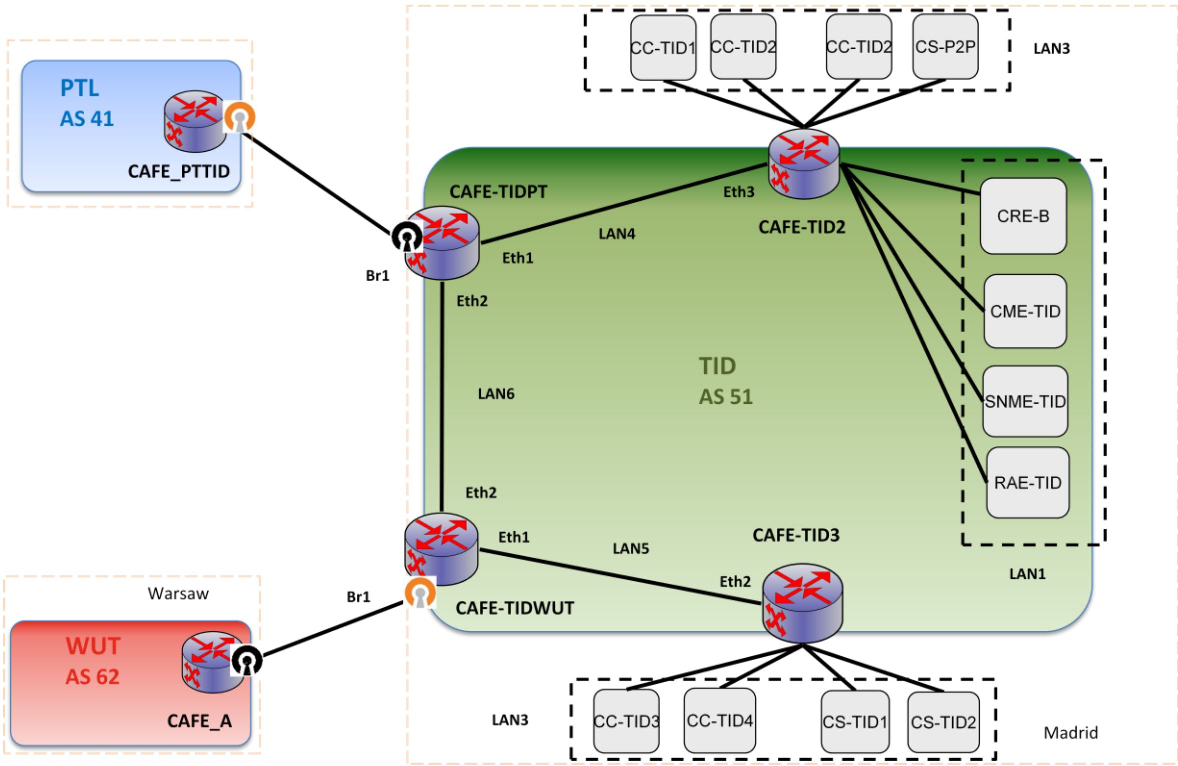


Figure 2: Final Layout of the TID' side of the Federated Testbed

All the elements in the figure have been configured in dual-stack so that they expose IPv6 and IPv4 address in all their interfaces. The complete list of IP/MAC addresses and interfaces can be found in Annex A.1.1. As a general rule of thumb, COMET SW is launched only on IPv4 mode or IPv6 mode, but not both at the same time, apart from some exceptions as the RAE which runs in dual-stack.

More interesting is the distribution of addressing according to internal LANs, which will be used when configuring the routing information among testbeds. The net addressing schema used in TID is as reflected in the following table:

LAN	Meaning	IPV4	IPv6
LAN1	COMET Servers	10.95.51.0/27	:2A02:9008:0:1911/64
LAN2	Northern Clients and Servers	10.95.51.32/28	2A02:9008:0:1912/64
LAN3	Southern Clients and Servers	10.95.51.48/28	:2A02:9008:0:1913/64
LAN4	CAFE-TIDPT/ CAFE-TID2	10.95.51.64/30	2A02:9008:0:1914/64
LAN5	CAFE-TIDWUT/CAFE-TID3	10.95.51.68/3	2A02:9008:0:1915/64
LAN6	CAFE-TIDPT/CAFETID-WUT	10.95.51.108/30	2A02:9008:0:1916/64
LAN8	private subnet for CAFETID-PT	10.95.51.80/30	2A02:9008:0:1918/6
LAN9	private subnet for CAFETID-WUT	10.95.51.84/30	2A02:9008:0:1919/64

Table 1: Subnet Addressing in TID testbed

All the machines in the testbed are Ubuntu LTE 10.04, except from the CCs and the CS-P2P, which have Windows 7 installed.

In short, the machines composing TID local testbed can be grouped in the following sets, according to functionality:

- Two Border CAFEs, CAFE-TIDWUT and CAFE-TIDPT, managing traffic going to and from the neighboring ISPs, AS 62 at WUT and AS 41 at PTL, respectively.
  - Border Cafes are controlled and configured by COMET elements through their private IP addresses at LAN9 and LAN8 respectively.
- Two Edge CAFE, CAFE-TID2 and CAFE-TID3 managing incoming and outgoing traffic for the two set of servers defined in TID Local Tested.
  - Edge Cafes are controlled and configured by COMET elements through their IP addresses at LAN2 and 3 respectively.
- Four Content Clients, CC-TID1, CC-TID2, CC-TID3 and CC-TID4, for launching Content Name resolution and Content Retrieval. The Content Client have Firefox, VLC and µTorrent installed
  - COMET Content Client SW has been configured to send queries to CME at port 9091, either to CME IPv6 or IPv4 address.
- Three Content Servers, CS-TID2, CS-TID3 and CS-TID4, that can act both as Streaming Servers, implemented with VLC or VoD servers, using Apache.
  - VLCs streaming servers listen at port 8080.
  - Apache are configured to use port 80.
- One Content server for P2P content distribution, CS-P2P, using µTorrent in tracker mode
  - Tracking port is 23343
- One CME, CME-TID, for accepting resolution queries from the CC and returning the server data after resolution and decision process. CME configuration will be detailed in following sections, but here it suffices to comment that:
  - Queries from CC are received at port 9091
  - Queries from RAE are received at port 9090
  - Queries from remote CMEs are received at port 9092
- One RAE, RAE-TID, for composing the path map of the federated testbed with the information provided by neighboring RAEs
- One SNME, SNME-TID, for collecting status information from CSs and Edge CAFEs, than can be later used by the CME in the decision process
  - SNME listen for CME queries at port 8888
- One CRE, CRE-TID, for storing Content Records for the following ISP of the COMET federated testbed
  - Contents at TID, AS 51, that will be collectively named as tid1.es
  - Content at WUT, either AS 61, 62 or 63, which are collectively named as wut2.pl
  - The CRE listen for queries from CMEs, either local or remote, at the port 2461

The details of how these elements have been configured to work together are the topic of the following sections. Note that the configuration tasks have been organized according to functionality not by entity, in order to highlight the logical sequence of the configuration process.

3.2.2 Forwarding Key Configuration

When a CAFE receives a COMET packet, the header will have a list of forwarding keys. These lists of keys are translated by the CAFE into the Ethernet interface the packet will have to leave the CAFE for the next hop. Taking into account that the direction of the intercepted packet is from server to client, the following keys have to be defined in the testbed:

CAFE	Destination CAFE	Key
CAFE-TID2	CAFE-TID2	0xfa
CAFE-TID2	CAFE-TIDpt	0xff
CAFE-TIDpt	CAFE-TIDwut	0xfe
CAFE-TIDpt	CAFE-TID2	0xff
CAFE-TIDpt	cafe-pttid	0xfc
CAFE-TIDwut	cafe_a	0xfc
CAFE-TIDwut	CAFE-TID3	0xfd
CAFE-TIDwut	CAFE-TIDpt	0xfe
CAFE-TID3	CAFE-TID3	0xfb
CAFE-TID3	CAFE-TIDwut	0xfd

Table 2: Forwarding Keys defined in TID Testbed

The complete definition for the forwarding keys can be found in Annex A.1.2

3.2.3 Subnets Managed by CAFEs

CME needs to know which CSs are managed by which CAFE, so that it can configure the right CAFE when a CS has been chosen as content source for a CC resolution query. In TID testbed, there are only two possibilities

- CAFE-TID2 will manage all the CSs deployed at LAN2
- CAFE-TID3 will manage all the CSs deployed at LAN3

Full configuration can be found at A.1.3.

3.2.4 Path Configuration in RAE

The configuration of the RAE element deployed at RAE_TID is twofold. First, the QoS and CoS for internal paths have to be defined, and then the paths to neighbouring domains, in TID's case to WUT (AS 62) and PTL (AS 41). Note that when referring to paths, it is always implied that packets are transferred from the CS to the CC, not the opposite. Besides, when talking about QoS for path, three different parameters are involved:

- IPTD or maximum packet delay in link.
- IPLR or packet loss ratio.
- Bandwidth.

In this section, the explanation will be restricted to internal paths, while in section 3.5 the path definition for linking to neighboring domains will be explained.

Prior to path configuration the QoS targets that will guide CME in path have to be defined. This means that for each CoS (Pr and BtBE) and each CoS Parameter, two values have to be defined:

- Aspiration Value: How important this parameter will be in the path selection.
- Reservation Value: A measure of how much of the parameter can be booked for a single flow (the lower, the greater).

These values are summarized in Table 3.

QoS Parameter	CoS	Aspiration Level	Reservation Level
IPTD	Pr	0.15	0.4
IPTD	BtBE	0.500	1
IPLR	Pr	0.000001	0,0001
IPLR	BtBE	0.00001	0.001
BW	Pr	10000000	4000000
BW	BtBE	4000000	1000000

Table 3: Aspiration and Reservation Levels according to CoS in TID CME

Internally, the following paths have to be characterized inside TID testbed:

- From CAFE-TID2 to CAFE-TID3 (Internal)
- from CAFE-TID3 to CAFE-TID2 (Internal)
- FromCAFE-TID2 to CAFE -TIDPT (Outgoing to AS 41)
- From CAFE-TIDPT to CAFE -TID2 (Incoming from AS 41)
- From CAFE-TID3 to CAFE -TIDPT (Outgoing to AS 41)
- From CAFE-TIDPT to CAFE -TID3 (Incoming from AS 41)
- From CAFE-TID2 to CAFE -TIDWUT (Outgoing to AS 62)
- From CAFE-TIDWUT to CAFE -TID2 (Incoming from AS 62)
- From CAFE-TID3 to CAFE -TIDWUT (Outgoing to AS 62)
- From CAFE-TIDWUT to CAFE -TID3 (Incoming from AS 62)
- Inside CAFE-TID3
- Inside CAFE-TID2

For simplicity's sake, the paths inside TID have been configured as symmetrical, the only difference being the values assigned for each CoS. Therefore for Pr:

- BW is 20Mb
- IPTD is 0.001
- IPLR is 0.000001

While for BtBE:

- BW is 20Mb
- IPTD is 0.0005
- IPLR is 0.000001

The only discrepancy is for those paths linking CS/CC in the same subnet, which are slightly better. Therefore for Pr:

- BW is 50Mb
- IPTD is 0.001
- IPLR is 0.000001

While for BtBE:

- BW is 50Mb
- IPTD is 0.0005
- IPLR is 0.0000001

The full configuration can be found at A.1.4.

3.2.5 Peering CAFEs/Keys Mappings

To enable the CME to configure a server's edge CAFE with the right sequence of keys, all the paths connecting a CS inside TID with a CC inside TID or a CS with a testbed exit point or a testbed entry point with a CC have to be configured.

Note that keys for connecting two neighboring ISPs will be described in section 3.5.

The defined key sequences are therefore:

- Inside TID Testbed:
  - Inside LAN2. Key sequence (in reverse order) is 0xfa
  - From LAN2 to LAN3. Key sequence (in reverse order) is 0xfd,0xfe,0xff
  - From LAN3 to LAN2. Key sequence (in reverse order) is 0xff,0xfe,0xfd
  - Inside LAN3. Key sequence (in reverse order) is 0xfa
- Exiting TID:
  - From LAN2 to PTL. Key sequence (in reverse order) is 0xfd.
  - From LAN2 to WUT. Key sequence (in reverse order) is 0xfd,0xfe.
  - From LAN3 to WUT. Key sequence (in reverse order) is 0xff.
  - From LAN3 to PTL. Key sequence (in reverse order) is 0xff,0xfe.
- Entering TID:
  - From PTL to LAN2. Key sequence (in reverse order) is 0xff.
  - From PTL to LAN3. Key sequence (in reverse order) is 0xfe,0xfd.
  - From WUT to LAN2. Key sequence (in reverse order) is 0xfe, 0xff.
  - From WUT to LAN3. Key sequence (in reverse order) is 0xfd.
- Crossing TID:
  - From PTL to WUT. Key sequence (in reverse order) is 0xfe
  - From WUT to PTL. Key sequence (in reverse order) is 0xfe

These paths have to be defined both for CoS Pr and BtBE, the difference being that they are assigned different capacity. As a rule of thumb, inside TID testbed:

- Pr paths are assigned 1GB
- BtBE paths are assigned 500MB

Detailed configuration for paths can be found at Annex A.1.6.

3.2.6 CC/Comet CoS Mappings

For a proper working of the COMET system, each CC has to be assigned a CoS that will be used by the CME to choose the appropriate Content Sources in the Content Records and the optimal paths among the choice offered by the CRE.

Basically the mapping in TID testbed is as follows:

- CC-TID1 is BE.
- CC-TID4 is BtBE
- CC-TID2 and CC-TID3 are Pr

The mappings for IPv6 and IPv4 are described in the Annex A.1.7.

3.2.7 CS/Comet CoS Mappings

For a proper working of the COMET system, whenever a Content Name is created, its Content Sources have to be assigned a CoS, so that only the right type of CC can access the servers included in the Content Source.

Although the same server can be assigned different CoS in different Content Sources, a simple set of rules has been established inside TID:

- CS-TID2 and CS-TID3 will normally act as Pr Servers.

- CS-TID4 is both BtBE and BE.
- CS-P2P is always BE
- For testing purposes, CS-TID2 can act as a BtBE, so that BtBE servers are deployed both in the northern and southern branches of TID testbed.

3.2.8 Content Names and Content Sources

The CRE deployed at CRE-TID hosts content names for both TID and WUT testbeds, identified respectively as domains name tid1.es and wut2.pl. Only the Content Names hosted in TID will be described here, while WUT ones are explained in Section 3.3.

Basically the content names defined in CRE-TID for the domain tid1.es are:

- tid1.es/cleoP2P. BE Content distributed via a P2P overlay managed by a torrent tracker located at CS-P2P.
- tid1.es/cleoSD. Pr Content distributed by HTTP streaming servers located at CS-TID2 and CS-TID3.
- tid1.es/cleoSD_BE. BE Content distributed by a HTTP streaming server located at CS-TID4
- tid1.es/cleoVOD. Pr Content distributed by VoD servers (Apache) located at CS-TID2 and CS-TID3
- tid1.es/cleoVOD_BE. BE Content distributed by a VoDserver (Apache) located at CS-TID4
- tid1.es/cleoMIX. Content Name comprising a PR HTTP streaming server at CS-TID2 and BE torrent tracker at CS-TID4
- tid1.es/cleoSD_BtBE. BtBE Content distributed by HTTP streaming servers located at CS-TID2 and CS-TID4.
- tid1.es/cleoVOD_BtBE. BtBE Content distributed by VoD servers (Apache) located at CS-TID2 and CS-TID3

More complex content names have been defined for the use case demonstrations and will be explained in section 3.7.

The full description of the Contents Names and its associated content Sources both for IPv4 and IPv6 mode can be found at Annex A.1.8.

3.2.9 Decision algorithm Tuning

The final step is the tuning of the decision algorithm. Basically there are six parameters that can be modified:

- Number of candidates, or how many paths/servers will be taken into account for decision
- Server Load
- Path Length. Maximum hops tolerated in a path
- IPTD. Maximum Packet Delay
- IPLR. Packet Loss Ratio
- BW.

The parameters can be configured using three different criteria:

- It can be strict or tolerant. Strict means that those paths/servers not matching the expectations will be discarded
- Reservation. Maximum value of the parameter the decision will consider acceptable. For some parameters this value cannot be selected:
  - Server Load has reservation preset to 3
  - IPTD, IPLR and BW obtain their reservation values from RAE
- Aspiration. Importance of the parameter in the decision algorithm. The lowest it is, the more important the parameter in the selection algorithm.

Parameter	Value
Candidates	100
Server Load	Strict Reservation: Not Modifiable Aspiration: 0,33
Path Length	Tolerant Reservation: 10.0 Aspiration: 0,99
IPTD	Tolerant Reservation: Not Modifiable Aspiration: 0,99
IPLR	Tolerant Reservation: Not Modifiable Aspiration: 0,01
BW	Tolerant Reservation: Not Modifiable Aspiration: Not Modifiable

Table 4: Decision Algorithm Parameters in TID CME

TID testbed has been thus configured to be intolerant for Server Load (CSs in high status will never be chosen as candidates) and to use IPLR as the most important parameter for the path ranking.

3.3 *WUT Local Testbed*

This section describes the layout of WUT local testbed. Some references to federated testbed interconnection are included for text consistency (federated testbed interconnection is described in detail in section 3.5).

3.3.1 Local Testbed Layout

Figure 3 depicts the layout of WUT local Testbed. All elements are referred by name. Since the Ethernet interfaces are required for configuration of CAFE forwarding keys, they are labelled in the figure.

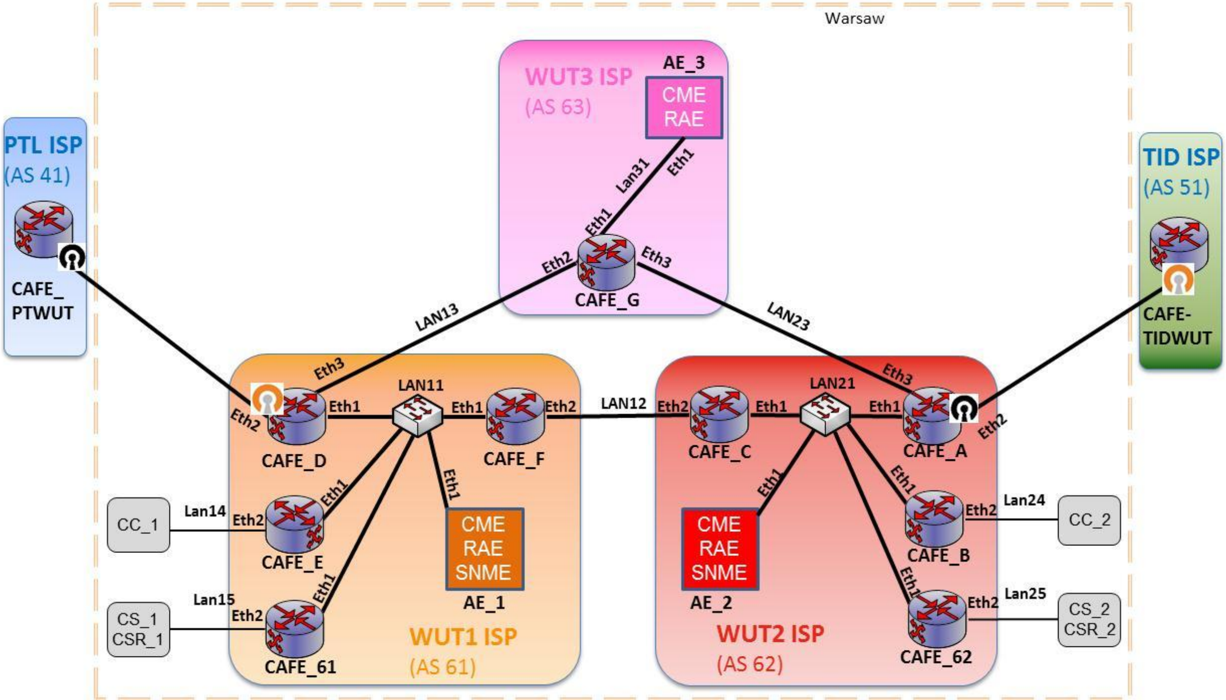


Figure 3: Final Layout of the WUT Local Testbed

All elements in the WUT testbed are configured with both IPv6 and IPv4 address. The complete list of IP/MAC addresses and interfaces can be found in Annex A.2.1. In general, the COMET system is launched only on IPv4 mode or IPv6 mode, but not both at the same time (apart from some exceptions as the RAE which runs in a dual-stack mode). All interfaces have assigned global IPv4 addresses. In case of IPv6, the link local IPv6 addresses are assigned by auto-configuration function for all interfaces. Moreover, we assigned global IPv6 addresses to interfaces facing Content Clients, Content Servers, COMET servers and CAFEs to enable communication with these elements.

The addressing schema used in WUT testbed is presented in Table 5. These addresses are used in configuration of BGP-4 and OSPF routing protocols. The detailed network diagram of WUT testbed, containing all configured IP (IPv4 and IPv6) addresses can be found in Annex A.2.1

LAN	Meaning	IPV4	IPv6
LAN 11	COMET servers and CAFEs in AS61	10.203.61.224/28	2001:67c:24cc:31c1::/64
LAN 12	Interconnection between AS61 and AS62	10.203.61.246/31	(link-local addresses based on auto-configuration)

LAN	Meaning	IPV4	IPv6
LAN 13	Interconnection between AS61 and AS63	10.203.63.246/31	link-local addresses based on auto-configuration)
LAN 14	Content Clients in AS 61	10.203.61.0/27	2001:67c:24cc:31c2::/64
LAN 15	Content Servers in AS 61	10.203.62.240/30	2001:67c:24cc:31c3::/64
LAN 21	COMET servers and CAFEs in AS62	10.203.62.224/28	2001:67c:24cc:31d1::cafe:a/64
LAN 23	Interconnection between AS62 and AS63	10.203.62.246/31	link-local addresses based on auto-configuration)
LAN 24	Content Clients in AS 62	10.203.62.0/27	2001:67c:24cc:31d2::/64
LAN 25	Content Servers in AS 62	10.203.62.240/30	2001:67c:24cc:31d3::/64
LAN 31	COMET servers and CAFEs in AS63	10.203.63.224/28	2001:67c:24cc:31e1::/64

Table 5: Subnet Addressing in WUT local testbed

All CAFE machines in WUT local testbed are based on Linux TinyCore 4.0.1 operating system, whereas all COMET servers are based on Linux Debian 3.1.5-1. The Content Clients are Windows terminals (WinXP or Win7).

As a general rule, the machines composing each domain of WUT local testbed can be grouped in the following sets:

<u>For AS 61:</u>

- Two Border CAFEs, CAFE_D and CAFE_F, managing traffic going to and from the neighbouring ISPs. Specifically, CAFE_D manages traffic to/from domain 41 at PTL and domain 63 at WUT, whereas CAFE_F manages traffic to/from domain 61 at WUT.
  - Border Cafes are controlled and configured by COMET elements through their IP addresses at LAN11.
- Two Edge CAFEs, CAFE_61 and CAFE_E, managing traffic incoming and outgoing from servers and content clients located in domain 61.
  - Edge CAFEs are controlled and configured by COMET elements through their IP addresses at LAN11.
- One Content Clients, CC_1 for launching Content Name resolution and Content Retrieval. The Content Client have installed Firefox, VLC and µTorrent applications.
  - COMET Content Client software has been configured to send queries to the CME at port 9091, either to CME IPv6 or IPv4 address.
- One Content Servers (CS_1, CSR_1), that can act both as Streaming Servers implemented with tornado server and Content Streaming Relay.
  - CS_1 listens at port 8899.
  - CSR_1 listens at port 8769.
- One CME for accepting requests from the CC and returning the server data after resolution and decision process. CME configuration is detailed in following sections, but here it suffices to comment that:
  - Queries from CC are received at port 9091.
  - Queries from RAE are received at port 9090.
  - Queries from remote CMEs are received at port 9092.
- One RAE for composing the path map of the federated testbed with the information provided by neighbouring RAEs.

- One SNME for collecting status information from CSs and Edge CAFEs, than can be later used by the CME in the decision process.
    - SNME listens for CME queries at port 8888.

for AS 62:

- Two Border CAFEs, CAFE_A and CAFE_C, managing traffic going to and from the neighbouring ISPs. Specifically, CAFE_A manages traffic to/from domain 51 at TID and domain 63 at WUT, whereas CAFE_C manages traffic to/from domain 62 at WUT.
    - Border Cafes are controlled and configured by COMET elements through their IP addresses at LAN21.
- Two Edge CAFEs, CAFE_62 and CAFE_B, managing traffic incoming and outgoing from servers and content clients located in domain 62.
    - Edge CAFEs are controlled and configured by COMET elements through their IP addresses at LAN21.
- One Content Clients, CC_2 for launching Content Name resolution and Content Retrieval. The Content Client has installed Firefox, VLC and µTorrent applications.
    - COMET Content Client software has been configured to send queries to CME at port 9091, either to CME IPv6 or IPv4 address.
- One Content Servers (CS_1, CSR_1), that can act both as Streaming Servers implemented with tornado server and Content Streaming Relay.
    - CS_2 listens at port 8899.
    - CSR_2 listens at port 8769.
- One CME for accepting content requests from the CC and returning the server data after resolution and decision process. CME configuration is detailed in following sections, but here it suffices to comment that:
    - Queries from CC are received at port 9091.
    - Queries from RAE are received at port 9090.
    - Queries from remote CMEs are received at port 9092.
- One RAE for composing the path map of the federated testbed with the information provided by neighbouring RAEs.
- One SNME for collecting status information from CSs and Edge CAFEs, than can be later used by the CME in the decision process.
    - SNME listens for CME queries at port 8888.

for AS 63:

- One Border CAFE, CAFE_G, managing traffic going to and from the neighbouring ISPs. Specifically, CAFE_G manages traffic to/from domain 61 and domain 63 at WUT.
    - CAFE_G is controlled and configured by COMET elements through its IP addresses at LAN31.
- One CME for path management. CME configuration is detailed in following sections, but here it suffices to comment that:
    - Queries from RAE are received at port 9090.
    - Queries from remote CMEs are received at port 9092.
- One RAE for composing the path map of the federated testbed with the information provided by neighbouring RAEs.

The details of how these elements are configured to work together are presented in the following sections. Note that the description is presented according to functionality not by entity, in order to highlight the logical sequence of the process.

3.3.2 Forwarding Key Configuration

When a CAFE receives a COMET packet, its header contains a list of forwarding keys. These keys are translated by the CAFE into the Ethernet interfaces which are used to forward the packet to the next hop. The CAFEs in WUT local testbed have been configured accordingly to Table 6. Notice that edge cafes facing content clients (i.e. CAFE_B and CAFE_E) are not configured with keys because no content traffic is generated by content clients.

CAFE	Destination CAFE	Key
CAFE_A	CAFE_B	0xeb
CAFE_A	CAFE_C	0xec
CAFE_A	CAFE_62	0xe6
CAFE_A	CAFE_G	0x63
CAFE_A	CAFE_TIDWUT	0x51
CAFE_C	CAFE_A	0xea
CAFE_C	CAFE_B	0xeb
CAFE_C	CAFE_62	0xe6
CAFE_C	CAFE_F	0x61
CAFE_62	CAFE_A	0xea
CAFE_62	CAFE_B	0xeb
CAFE_62	CAFE_C	0xec
CAFE_D	CAFE_F	0xef
CAFE_D	CAFE_E	0xee
CAFE_D	CAFE_61	0xe6
CAFE_D	CAFE_G	0x63
CAFE_D	CAFE_PTWUT	0x41
CAFE_F	CAFE_D	0xed
CAFE_F	CAFE_E	0xee
CAFE_F	CAFE_61	0x61
CAFE_F	CAFE_C	0x62
CAFE_61	CAFE_D	0xed
CAFE_61	CAFE_E	0xee
CAFE_61	CAFE_F	0xef
CAFE_G	CAFE_A	0x62
CAFE_G	CAFE_D	0x61

Table 6: Forwarding keys configured in WUT local testbed

The complete definition for the forwarding keys can be found in Annex A.2.2

3.3.3 Subnets Managed by CAFEs

The CME needs to know which CSs are managed by which particular CAFE in order to configure the right CAFE when this CS has been chosen as content source for a CC resolution query. In WUT local testbed, there are only two possibilities:

- CAFE-61 manages all the CSs deployed at LAN15,
- CAFE-62 manages all the CSs deployed at LAN25.

Full configuration can be found at A.2.3.

3.3.4 Path Configuration in RAE

The configuration of the RAEs elements in WUT local domain consists of two parts: a) definition of the Classes of Services with associated QoS parameters that are supported on internal paths inside domain, and b) definition of the paths to neighbouring domains (in PTL, TID and WUT local domains). Note that when referring to paths, it is always implied that packets are transferred from the content server to the content client. Besides when talking about QoS for path, three parameters are considered:

- IPTD – maximum value of IP Packet Transfer Delay
- IPLR – maximum value of IP Packet Loss Ratio

- Bandwidth.

In this section, the explanation is restricted to internal paths inside domains in WUT local testbed, while section 3.5defines the paths linking to neighbouring domains.

The following paths have to be characterized inside WUT testbed:

For AS 61:

External paths:

- From CAFE_D to CAFE_G (Outgoing to AS 63)
- From CAFE_G to CAFE_D (Incoming from AS 63)
- From CAFE_F to CAFE_C (Outgoing to AS 61)
- From CAFE_C to CAFE_F (Incoming from AS 62)
- From CAFE_D to CAFE_PTLWUT (Outgoing to AS 41)
- From CAFE_PTLWUT to CAFE_D (Incoming from AS 41)

Internal paths:

- From CAFE_61 to CAFE_D
- From CAFE_61 to CAFE_F
- From CAFE_61 to CAFE_E
- From CAFE_D to CAFE_D
- From CAFE_D to CAFE_E
- From CAFE_D to CAFE_F
- From CAFE_D to CAFE_61
- From CAFE_F to CAFE_D
- From CAFE_F to CAFE_E
- From CAFE_F to CAFE_61

For AS 62:

External paths:

- From CAFE_A to CAFE_G (Outgoing to AS 63)
- From CAFE_G to CAFE_A (Incoming from AS 63)
- From CAFE_F to CAFE_C (Incoming from AS 61)
- From CAFE_C to CAFE_F (Outgoing to AS 61)
- From CAFE_A to CAFE_TIDWUT (Outgoing to AS 51)
- From CAFE_TIDWUT to CAFE_A (Incoming from AS 51)

Internal paths:

- From CAFE_62 to CAFE_A
- From CAFE_62 to CAFE_B
- From CAFE_62 to CAFE_C
- From CAFE_A to CAFE_A
- From CAFE_A to CAFE_B
- From CAFE_A to CAFE_C
- From CAFE_A to CAFE_62
- From CAFE_C to CAFE_A
- From CAFE_C to CAFE_B
- From CAFE_C to CAFE_62

For AS 63:

External paths:

- From CAFE_G to CAFE_D (Outgoing to AS 61)
- From CAFE_D to CAFE_G (Incoming from AS 61)
- From CAFE_G to CAFE_A (Outgoing to AS 62)
- From CAFE_A to CAFE_G (Incoming from AS 62)

Internal paths:

- From CAFE_G to CAFE_G

For simplicity's reasons in domains of WUT local testbed, all paths apart from paths linking CS/CC are configured as symmetrical with following values assigned for each CoS:

Pr:

- BW is 100Mbps
- IPTD is 0.0005 s
- IPLR is 0.000001

BtBE:

- BW is 100Mbps
- IPTD is 0.001 s
- IPLR is 0.000001

Paths linking CS/CC in the same AS, are configured with slightly better QoS values. Therefore for Pr:

- BW is 1000Mbps
- IPTD is 0.0005 s
- IPLR is 0.000001

while for BtBE

- BW is 1000Mbps
- IPTD is 0.0005 s
- IPLR is 0.0000001

The detailed configuration can be found at A.2.4

3.3.5 Peering CAFEs/Keys Mappings

The CME located in each domain is responsible for configuring a server's edge CAFE with the right sequence of forwarding keys. Therefore, we need to configure list of keys for all paths connecting: a) a CS inside AS with a CC inside AS or b) a CS with a AS egress point or c) a AS ingress point with a CC have to be configured.

Note that keys for connecting two neighbouring ISPs will be described in section 3.5.

The defined key sequences are therefore

For AS61:

- Inside AS 61
    - From LAN15 to LAN14. Key sequence (in reverse order) is 0xee.
- Exiting AS 61
    - From LAN15 to PTL (AS 41). Key sequence (in reverse order) is 0xed.
    - From LAN15 to AS 62. Key sequence (in reverse order) is 0xef.
    - From LAN15 to AS 63. Key sequence (in reverse order) is 0x63.
- Entering AS 61
    - From PTL (AS 41) to LAN15. Key sequence (in reverse order) is 0xe6.
    - From PTL (AS 41) to LAN14. Key sequence (in reverse order) is 0xee.
    - From AS 62 to LAN15. Key sequence (in reverse order) is 0x61.
    - From AS 62 to LAN14. Key sequence (in reverse order) is 0xee
    - From AS 63 to LAN15. Key sequence (in reverse order) is 0xe6.
    - From A S63 to LAN14. Key sequence (in reverse order) is 0xee
- Crossing AS 61
    - From PTL (AS 41) to AS 62 Key sequence (in reverse order) is 0xef

- From PTL (AS 41) to AS 63 Key sequence (in reverse order) is null
- From AS 62 to PTL (AS 41). Key sequence (in reverse order) is 0xed
- From AS 62 to AS 63. Key sequence (in reverse order) is 0xed
- From AS 63 to PTL (AS 41). Key sequence (in reverse order) is null
- From AS 63 to AS 62. Key sequence (in reverse order) is 0xef

For AS 62:

- Inside AS 62
  - From LAN25 to LAN24. Key sequence (in reverse order) is 0xee
- Exiting AS 62
  - From LAN25 to TID (AS 51). Key sequence (in reverse order) is 0xea.
  - From LAN25 to AS 61. Key sequence (in reverse order) is 0xec.
  - From LAN25 to AS 63. Key sequence (in reverse order) is 0x63.
- Entering AS 62
  - From TID (AS 51) to LAN25. Key sequence (in reverse order) is 0xe6.
  - From TID (AS 51) to LAN24. Key sequence (in reverse order) is 0xeb.
  - From AS 61 to LAN25. Key sequence (in reverse order) is 0xe6.
  - From AS 61 to LAN24. Key sequence (in reverse order) is 0xeb
  - From AS 63 to LAN15. Key sequence (in reverse order) is 0xe6.
  - From AS 63 to LAN24. Key sequence (in reverse order) is 0xeb
- Crossing AS 62
  - From TID (AS 51) to AS 61 Key sequence (in reverse order) is 0xec
  - From TID (AS 51) to AS 63 Key sequence (in reverse order) is null
  - From AS 61 to TID (AS 51). Key sequence (in reverse order) is 0xea
  - From AS 61 to AS 63. Key sequence (in reverse order) is 0xea
  - From AS 63 to TID (AS 51). Key sequence (in reverse order) is null
  - From AS 63 to AS 61. Key sequence (in reverse order) is 0xec

For AS 63:

- Crossing AS63
  - From AS 62 to AS 61. Key sequence (in reverse order) is null
  - From AS 61 to AS 62. Key sequence (in reverse order) is null

Detailed configuration for Paths can be found at Annex A.2.6.

3.3.6 CC/Comet CoS Mappings

Each CC is subscribed to a particular CoS. During the resolution process, the CME chooses the appropriate Content Sources and paths suitable to subscribed CoS. Basically the mapping in WUT testbed is as follows

- CC_1 (located in AS61) is Pr.
- CC_2 (located in AS61) is Pr.

Moreover content client assigned to other classes (BE or BTBE) may be added to AS61 and AS62 if needed. The mappings for IPv6 and IPv4 are described in the Annex A.2.7.

3.3.7 CS/Comet CoS Mappings

For a proper working of the COMET server, whenever a Content Name is created, their Content Sources have to be assigned a CoS, so that only the right type of CC can access the servers in the Content Source.

Although a same server can be assigned different CoS in different Content Sources, in WUT testbed all CSs (CS_1, CSR_1, CS_2 and CSR_2) are assigned to Pr service. However, the assignment can be modified following requirements of particular test.

3.3.8 Content Names and Content Sources

The WUT testbed uses the Authoritative CRE deployed in TID. The WUT domain name space covers all contents registered as wut2.pl

Basically the content names defined for the domain wut2.pl are:

- wut2.pl/cars_be. BE content in standard resolution distributed in VoD mode from both CS_1 and CS_2 servers
- wut2.pl/cars_pr. Pr content in standard resolution distributed inVoD mode from both CS_1 and CS_2 servers
- wut2.pl/sun_be. BE content in high resolution distributed in VoD mode from both CS_1 and CS_2 servers
- wut2.pl/sun_pr. Pr content in high resolution distributed in VoD mode from both CS_1 and CS_2 servers
- wut2.pl/match_be. BE content in standard resolution distributed as HTTP streaming from CSR_1 and CSR_2
- wut2.pl/match_pr. Pr content in standard resolution distributed as HTTP streaming from CSR_1 and CSR_2

The full description of the Contents Names and its associated content Sources both for IPv4 and IPv6 mode can be found at Annex A.1.8

3.3.9 Decision algorithm tuning

The basic set of decision parameters used in domain 61 and 62 are summarized in Table 7 and Table 8, respectively. The domain 63 does not use the decision algorithm because it is a transit domain without content clients.

Parameter	Value
Candidates	100
Server Load	Strict Reservation: Not Modifiable Aspiration: 0.9
Path Length	Strict Reservation: 5.0 Aspiration: 0.1
IPTD	Strict Reservation: Not Modifiable Aspiration: 0.75
IPLR	Strict Reservation: Not Modifiable Aspiration: 0.1
BW	Strict Reservation: Not Modifiable Aspiration: Not Modifiable

Table 7: Decision algorithm parameters in WUT domain 61

Parameter	Value
Candidates	100

Server Load	Strict Reservation: Not Modifiable Aspiration: 0.9
Path Length	Strict Reservation: 5.0 Aspiration: 0.1
IPTD	Strict Reservation: Not Modifiable Aspiration: 0.75
IPLR	Strict Reservation: Not Modifiable Aspiration: 0.1
BW	Strict Reservation: Not Modifiable Aspiration: Not Modifiable

Table 8: Decision algorithm parameters in WUT domain 62

3.4 PTL Local Testbed

This section describes the testbed configuration at the Primetel's site (PTL). Primetel's Local Testbed setup took place at Primetel's Limassol HQs Network Testbed. The complete Federated Testbed is illustrated Figure 2 of which this local testbed formed a part.

3.4.1 Local Testbed Layout

Figure 4 shows a layout of PTL's Testbed. IPv4/IPv6 address and MAC configuration are not shown but are addressed latter in the document. The figure shows the Ethernet interfaces used with a focus on the CAFEs which were used for tested interconnection.

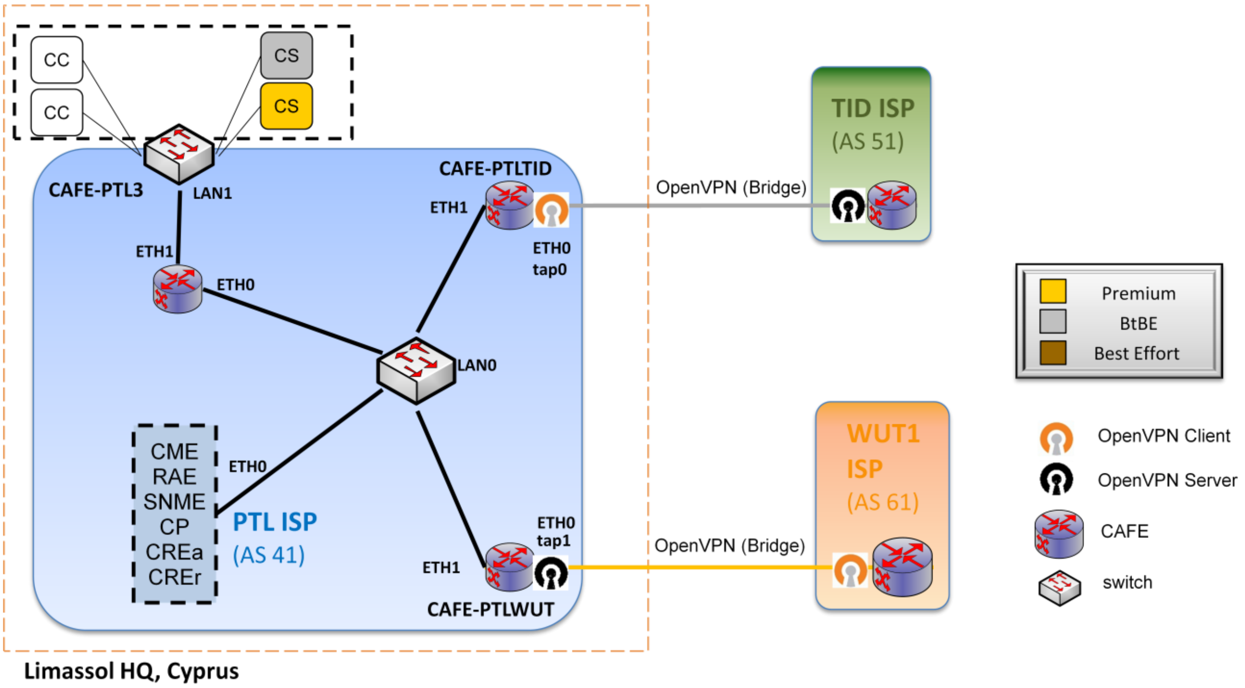


Figure 4: PTL's Local Testbed

All component interfaces in Figure 4 have been configured to support both IPv4 and IPv6. Note however that the testbed could be configured in either address system but not in an inter-working IPv6/IPv4 configuration. The complete list of IP/MAC addresses and interfaces can be found in.

Table 9 shows the IPv4 and IPv6 addressing used according to the testbed LANs used for the internal connectivity.

LAN	Serving	IPV4	IPv6
LAN1	CME, RAE, SNME, CP, CREa, CREr, $CAFE_{PTLTID}$, $CAFE_{PTLWUT}$	10.50.50.0/24	3105::/64
LAN2	CCs, CSs	101.101.101.0/24	3101::/64

Table 9: Subnet Addressing in PTL

The selection of operating systems, hardware resources and the associated software are summarised below, in Table 10.

Entity	OS	Apps	RAM	HDD
CME	Ubuntu	Apache Ant, Java Development Kit, .NET, MySQL Server	256MB	5GB
CRE	Ubuntu	Apache Ant, Java Development Kit	256MB	5GB
RAE	Ubuntu	Apache Ant, Java Development Kit, sqlite3, waf	256MB	5GB
SA	Ubuntu	Apache ANT, Java Development Kit, MySQL	256MB	5GB
CAFE forward/edge	Tiny CORE	Quagga, Python	128MB	0.5GB
CS-SMA	Ubuntu	Apache ANT, Apache Tomcat, Java Development Kit, MySQL, VLC	256MB	5GB
CP	Ubuntu	Apache ANT, Java Development Kit, MySQL	256MB	5GB
CC	Ubuntu	VLC, Mozilla Forefox, µTorrent	256MB	5GB

Table 10: Resource allocation PTL local testbed

Information relevant to the abovementioned entities are summarised below:

- Two border CAFEs where used namely $CAFE_{PTLTID}$, $CAFE_{PTLWUT}$, for routing traffic to the neighboring ISPs, AS 51 at TID AS 61 at WUT.
  - Border Cafes are controlled and configured by COMET elements through their private IP addresses at LAN0.
- $CAFE_{PTL3}$edge CAFE manages any incoming and outgoing traffic destined or being sent by any of Content Servers or Content Clients at LAN1.
  - $CAFE_{PTL3}$ is controlled and configured by COMET elements through their IP address at LAN1.
- 1 Content Client, CC_{PTL1}, is available at PTL local testbed containing Firefox, VLC and µTorrent for performing Content consumptions.
  - Configured to send queries at port 9091 on the CME.
- 2 Content Servers, CS_{PTL1} and CS_{PTL2} are equipped with Apache and VLC for enabling VoD and streaming.
  - VLC Server listens on port 8080.
  - Apache listens on port 8008.
- 1 CME, namely CME_{PTL} is set up, accepting queries from the CC and in turn returning the content address after the completion of all necessary processes.
  - Queries from CC are received at port 9091.
  - Queries from RAE are received at port 9090.
  - Queries from remote CMEs are received at port 9092.
- 1 RAE, RAE_{PTL} for composing the path map and associated network information of the federated testbed for all available paths using local and remote RAE information.
- 1 SNME, $SNME_{PTL}$ for collecting memory/network/process overall load status from CSs and Edge CAFEs required by CME's decision process
  - Configured to listen for CME queries at port 8888.
- 1 CRE, CRE_{PTL} for storing Content Records for local ISP of the COMET federated testbed i.e. PTL, AS 41, (referred to as ptl3.cy).
  - The CRE listen for queries from CMEs (local and remote) at the port 2461.

Configurations regarding the interworking of the above components are described in the sections to follow. Organisation of these has been established logically according to functionality.

3.4.2 Forwarding Key Configuration

A COMET packet destined to a CAFE contains a list of forwarding keys. These keys are translated by the CAFE into the Ethernet interface that the packet has to leave the CAFE for the hop to follow. The following keys have been defined at the PTL CAFEs.

CAFE	Destination CAFE	Key
cafe-ptl3	cafe-pt l3	0x01
cafe-pt l3	cafe-ptlwut	0x21
cafe-pt l3	cafe-ptltid	0x31
cafe-ptlwut	cafe-pt l3	0x22
cafe-ptltid	cafe-pt l3	0x32
cafe-ptlwut	cafe-pttid	0x11
cafe-ptltid	cafe-pttid	0x12
cafe-ptlwut	cafe-d (wut)	0xb1
cafe-ptltid	CAFE-TIDpt (tid)	0xc1

Table 11: Forwarding Keys defined in the PTL Testbed

The complete definition for the forwarding keys can be found in Annex 3.4.2.

3.4.3 Subnets Managed by CAFEs

CMEs need to know which CSs are managed by which CAFEs, so that they configure the right CAFE when a CS has been chosen as content source for a CC resolution query. At the PTL testbed, there is only one:

- CAFE-PTL3 managing all the CSs deployed at LAN1.

3.4.4 Path Configuration in RAE

In the deployed RAE first the QoS and CoS for the internal paths is defined, and secondly the paths to the neighboring domains are defined, namely to WUT (AS 61) and TID (AS51). As mentioned earlier for defining the QoS three parameters need to be defined: IPTD or maximum packet delay in link, IPLR or packet loss ratio and Bandwidth. These are defined prior to path configuration. For each CoS Pr and BtBE, two values are then defined:

- Aspiration value, or how important this parameter will be in the path selection.
- Reservation value, or a measure of how much of the parameter can be booked for a single flow.

These values are summarized in the following table

QoS Parameter	CoS	Aspiration Level	Reservation Level
IPTD	Pr	0.15	0.4
IPTD	BtBE	0.500	1
IPLR	Pr	0.000001	0,0001
IPLR	BtBE	0.00001	0.001
BW	Pr	10000000	4000000
BW	BtBE	4000000	1000000

Table 12: Aspiration and Reservation Levels according to CoS in PTL CME

Internally, the following paths have to be characterized inside PTLs testbed:

- From CAFE-PTLWUT to CAFE-PTLTID (Internal).

- From CAFE-PTLTID to CAFE-PTLWUT (Internal).
- From CAFE-PTL3 to CAFE-PTLTID (Outgoing to AS51).
- From CAFE-PTL3 to CAFE-PTLWUT (Outgoing to AS61).
- From CAFE-PTLTID to CAFE-PTL3 (Incoming from AS51).
- From CAFE-PTLWUT to CAFE-PTL3 (Incoming from AS61).
- Inside CAFE-PTL3 (Internal).

The paths inside Primetel's Local Testbed have been configured as symmetrical with different values assigned for each CoS.

For Pr the values are:

- BW is 20Mb.
- IPTD is 0.001.
- IPLR is 0.000001.

While for BtBE

- BW is 20Mb.
- IPTD is 0.0005.
- IPLR is 0.000001.

However, the peer paths are asymmetrical in respect to loss ratio i.e. peer connection to AS51 has higher packet loss than peer connection to AS61. The values assigned to each peer at the equivalent CoS are as follows:

For AS 51 PR:

- BW is 20Mb
- IPTD is 0.01
- IPLR is 0.0001

For AS 51 BtBE:

- BW is 10Mb
- IPTD is 0.1
- IPLR is 0.001

For AS 61 PR:

- BW is 20Mb
- IPTD is 0.01
- IPLR is 0.000001

For AS 61 BtBE:

- BW is 10Mb
- IPTD is 0.1
- IPLR is 0.00001

3.4.5 Peering CAFEs/Keys Mappings

To enable the CME to configure a server-side edge CAFE with the right sequence of keys, all the paths connecting a CS with a CC inside PTL as well as a CS connected with a CC at a testbed exit/entry point are configured as follows:

- Inside PTL testbed
    - Inside LAN0. Key sequence is 0x01.
- Exiting PTL
    - From LAN0 (CAFE-PTLWUT) to WUT. Key sequence is 0xb1.
    - From LAN0 (CAFE-PTLTID) to TID. Key sequence is 0xc1.

  - From WUT to LAN0 (CAFE-PTLWUT). Key sequence is 0xb2.
  - From TID to LAN0 (CAFE-PTLTID). Key sequence is 0xc2.
- Entering PTL
  - From border cafe connected to WUT (CAFE-PTLWUT) to LAN1. Key sequence is 0x22.
  - From border cafe connected to TID (CAFE-PTLTID) to LAN1. Key sequence is 0x32.
  - From LAN1 to border cafe connected to WUT (CAFE-PTLWUT). Key sequence is 0x21.
  - From LAN1 to border cafe connected to TID (CAFE-PTLTID). Key sequence is 0x31.
- Crossing PTL
  - From WUT (CAFE-PTLWUT) to TID (CAFE-PTLTID). Key sequence is 0x11.
  - From WUT (CAFE-PTLWUT) to TID (CAFE-PTLTID).Key sequence is 0x12.

These paths have to be defined both for CoS Pr and BtBE, with however different capacity. In PTL these are:

- Pr paths are assigned 1GB.
- BtBE paths are assigned 500MB.

More details are included in Annex A.3.5.

3.4.6 CC/Comet CoS Mappings

In order for a CS to function properly, each CC has to be assigned a CoS that will be available to the CME for selecting the appropriate Content Sources in the Content Records and the suitable paths offered by the CRE.

Basically the mapping in PTL testbed is as follows:

- CC-PTL1 is PR.
- CC-PTL2 is BE

3.4.7 CS/Comet CoS Mappings

For the CS to function accordingly, when a Content Name is created, their Content Sources have to be assigned a CoS, so that the only right type of CC can access the servers in the Content Source.

Although the same server can be assigned different CoS in different Content Sources, a simple set of rules has been established in PTL:

- CS-PTL1 and CS-PTL2 act both as Pr and BTBE Servers

3.4.8 Content Names and Content Sources

The CRE hosted at PTL namely CREa, contains content names for PTL only identified asptl1.cy.

The content names defined in CREa for the ptl1.cy domain are:

- ptl3.cy /adSD. Pr Content distributed by HTTP via the use of VLC located at both CS-PTL1 and CS-PTL2.
- ptl3.cy/adSD_BtBE. BtBE Content distributed by HTTP via VLC located at both CS-PTL1 and CS-PTL2.
- ptl3.cy/adSD_BE. BE Content distributed by HTTP via VLC located at both CS-PTL1 and CS-PTL2.
- ptl3.cy/adVOD. PR Content distributed by VoD servers (Apache) located at both CS-PTL1 and CS-PTL2.
- ptl3.cy/adVOD_BtBE. BtBEContent distributed by VoD servers (Apache) located at both CS-PTL1 and CS-PTL2.
- ptl3.cy/adVOD_BE. BE Content distributed by VoD servers (Apache) located at both CS-PTL1 and CS-PTL2.

. The full description of the Contents Names and its associated content Sources both for IPv4 and IPv6 mode can be found at Annex A.1.8

3.4.9 Decision algorithm Tuning

As explained in 3.2.9 the final step in the setup is the tuning of the decision algorithm which requires modification of six parameters: 1) Number of candidates or how many paths/servers will be taken into account for decision, 2) Server Load, 3) Path Length. Maximum hops tolerated in a path, 4) IPTD, 5) IPLR and 6) Bandwidth.

For Primetel's testbed the configuration of the parameters are summarized in table 13 accordingly. The value of Strict or Tolerant relates to whether if the path/server is not matching the expected requirements will be discarded or not. Reservation is the maximum value of the parameter the decision will consider acceptable. Aspiration is the importance of the parameter in the decision algorithm with lower implying more important.

Parameter	Value
Candidates	100
Server Load	Strict Reservation: Not Modifiable Aspiration: 0,33
Path Length	Strict Reservation: 30.0 Aspiration: 0,33
IPTD	Tolerant Reservation: Not Modifiable Aspiration: 0,6
IPLR	Tolerant Reservation: Not Modifiable Aspiration: 0,66
BW	Tolerant Reservation: Not Modifiable Aspiration: Not Modifiable

Table 13: Decision Algorithm Parameters in PTL CME

3.5 *Testbed Interconnection*

The following tasks are needed when configuring the local testbeds for COMET federation testbed:

1. Testbeds are linked by OpenVPN bridges, where one testbed is the server and the other the client. In short:
    - PTL-TID. PTL client, TID server.
    - TID-WUT. TID client, WUT server.
    - WUT-PTL. WUT client, PTL server.
2. Quagga is used for ensuring basic IP routing and IP visibility.
3. Keys for forwarding traffic to neighboring border CAFEs have to be defined in the local border CAFE.
4. Paths for outgoing traffic, from the testbed to neighboring ISP have to be defined in the RAE.
5. Peering AS/CAFE Mappings, specifying which border CAFEs are in charge of both ends of a link between neighboring CAFEs, have to be defined in the CMEs.
6. Peering CAFEs/Keys Mappings,from the local border CAFEs to neighboring border CAFEs, have to be defined in the CMEs.
7. Root CRE has to be fed with information about authoritative CRE serving each domain.

Most of this tasks are similar to other already carried out for the local testbed configuration, so they will not explained again in detail.

3.5.1 Open VPN

OpenVPN [17] in an open source software that implements VPN (Virtual Private Network) service. It offers secure tunnels that carry IPv6/IPv4 packets or Ethernet frames respectively to the routed or bridged configuration. OpenVPN supports authentication based on pre-shared keys, certificates or username/password tuple.

COMET exploits OpenVPN in bridged configuration as it allows transfering custom protocols over Ethernet frames through VPN tunnel (VPN tunnel in routed configuration supports only IPv4/IPv6 datagrams to be carried through VPN tunnel). In COMET testbed authentication is based on Public Key Infrastructure (PKI) and certificates. Each OpenVPN server acts also as trusted Certification Authority and generates certificates and private keys for OpenVPN server and clients.

3.5.1.1 Open VPN in TID

To interconnect with the other testbeds, TID has set up two different tunnels:

- PTL-TID: In this case TID has set up the OpenVPN as server side:

  The OpenVPN server is working as a bridged VPN tunnel over UDP protocol.

  TID has deployed a gateway with a public IP address (130.206.214.70) and a UDP port (2011) to allow PTL reach the TID Server from the Internet. This gateway performs a NAT (Network Address Translation) to reach the internal IP address of the server (10.95.51.81).

  Also, TID has created several credentials to allow PTL OpenVPN clients connect to the server, this credentials are based on client authentication methods based on certificates.

  TID has setup the OpenVPN Server to assign IP addresses dynamically to the clients. Different routes to reach the networks are Also forwarded using OpenVPN Server.

  The OpenVPN configuration file is attached inAnnexA.1.9.1.

  The server has been installed in the machine “CAFE-TIDpt”

- WUT-TID: In this case TID has set up the OpenVPN software as client side:

  TID has setup the OpenVPN in client mode to connect to WUT OpenVPN Server with IP address 194.29.169.3 and port 2001.

The routes to reach the other testbeds are received using Quagga software (explained in Section 3.5.2)

The OpenVPN configuration file is attached in Annex.A.1.9.2

The client has been installed in the machine “CAFE-TIDwut”

3.5.1.2 Open VPN in PTL

PLT local testbed establishes connections with TID and WUT local testbeds by applying the following VPN tunnels:

- TID-PTL tunnel: PTL runs OpenVPN client. PTL is running OpenVPN in client mode in order to establish VPN connection with TID OpenVPN Server with IP address 130.206.214.70 at port 2011. OpenVPN is running on virtual machine CAFE_PTLTID. Furthermore, the OpenVPN virtual *tap* interface is bridged with interface eth0, which is connected to internet with public IP address 217.27.59.124. The OpenVPN configuration file can be found in AnnexA.3.8.1.
- PTL-WUT tunnel: PTL runs OpenVPN server. PTL is running OpenVPN in server mode in order to enable VPN connection from clients and basically connections from WUT. OpenVPN is running on virtual machine CAFE_PTLWUT and accepts OpenVPN connections at IP address 217.27.59.123 port 2011 (eth0) which is also bridged with the virtual *tap*. PTL credential/certificates were generated using OpenVPN and then provided to WUT. The OpenVPN configuration file can be found in AnnexA.3.8.2.

3.5.1.3 Open VPN in WUT

To interconnect WUT local testbed with the other testbeds (i.e. PT and TID local testbeds), WUT runs two VPN tunnels:

- WUT-TID tunnel: In this case, WUT runs OpenVPN server. WUT runs the OpenVPN in sewer mode at address 194.29.169.3 at port 2001 to connect with the TID OpenVPN client. The OpenVPN server software runs on the physical machine that hosts CAFE_A virtual machine. The OpenVPN virtual *tap* interface is bridged with interface eth2 interface of CAFE_A. The OpenVPN configuration file is attached in Annex A.2.9.2. WUT created several credentials to allow OpenVPN clients to connect to the server, this credentials are based on certificates. WUT has setup the OpenVPN Server to assign IP addresses dynamically to the clients. different routes to reach the networks are also forwarded using OpenVPN Server.
- PT-WUT tunnel: In this case, WUT runs OpenVPN client. WUT runs the OpenVPN in client mode to connect to PT OpenVPN Server with IP address 217.27.59.123 and at port 2011. The OpenVPN client software runs on the physical machine that hosts CAFE_D virtual machine. The OpenVPN virtual *tap* interface is bridged with interface eth2 interface of CAFE_D. The OpenVPN configuration file is attached in Annex A.2.9.1.

3.5.2 Quagga Configuration

Quagga is a network routing software, which provides implementation of BGP, OSPF, RIP routing protocols for UNIX platforms. Quagga is composed of several routing deamons (one per routing protocol) and core daemon called zebra, which acts as abstraction layer to Unix kernel. Quagga provides following routing daemons:

- ospfd - implementing OSPFv2,
- ripd - implementing RIP v1 and V2
- ospf6d - implementing OSPFv3 (IPv6)
- ripngd - implementing RIPng (IPv6)
- bgpd - implementing BGPv4+ (including address family support for multicast and IPv6).

Quagga may be configured via dedicated command line interpreter or via configuration files (one configuration file per one daemon).

3.5.2.1 Quagga in TID

Quagga has been set up on each CAFE inside the TID Federated Testbed:

- CAFE-TIDPT: The configured daemons are: Zebra, bgpd, ospfd and ospfd6. These daemons have the following functions:
- Create BGP Router for the CAFE-TIDWUT.
- Create the bgp neighbour with PT.
- Create the IPV6 neighbour.

The configuration is described in AnnexA.1.10.1

- CAFE-TIDWUT: The configured daemons are: Zebra, bgpd, ospfd and ospfd6.These daemons have the following functions:
- Create BGP Router for the CAFE-TIDPT.
- Create the bgp neighbour with PT.
- Create the IPV6 neighbour.

The configuration is described in Annex A.1.10.2.

- CAFE-TID2: The configured daemons are: Zebra, ospfd and ospfd6. These daemons have the following functions:
- Create interface BR to support Remote Access tunnel
- Create ospf router

The configuration is described in Annex A.1.10.4.

- CAFE-TID3: the configured daemons are: Zebra, bgpd, ospfd and ospfd6. These daemons have the following functions:
- Create ospf router

The configuration is described in Annex A.1.10.3.

3.5.2.2 Quagga in PTL

Quagga has been set up on each CAFE inside the PTL Federated Testbed:

- CAFE-PTLTID: The daemons configured are: Zebra, bgpd, ospfd and ospfd6. These daemons have the following functions:
- Create BGP Router for the CAFE-PTLTID
- Create the bgp neighbor with TID
- Create the IPV6 neighbor

The configuration is described in AnnexA.3.9.1.

- CAFE-PTLWUT: The daemons configured are: Zebra, bgpd, ospfd and ospfd6.These daemons have the following functions:
- Create BGP Router for the CAFE-PTLWUT
- Create the bgp neighbor with WUT
- Create the IPV6 neighbor

The configuration is described in Annex A.3.9.2.

- CAFE-PTL3: The daemons configured are: Zebra, ospfd and ospfd6. These daemons have the following functions:
    - Create ospf router and interconnecting access network with PTL/ISP network

The configuration is described in Annex A.3.9.3.

3.5.2.3 Quagga in WUT

Quagga runs on each CAFE inside the WUT Local Testbed and is responsible for inter and intra-domain routing.

AS61:

- CAFE_D: runs the following daemons: Zebra, bgpd, ospfd and ospfd6. These daemons perform the following functions:
- Create BGP IPv4 router for the CAFE_D
- Create BGP IPv6 router for the CAFE_D
- Create the BGP peer with AS 41 (PT)
- Create the BGP peer with AS 63 (WUT)
- Create OSPF IPv4 router
- Create OSPF IPv6 router

The configuration is described in Annex A.1.10.1.

- CAFE_D: runs the following daemons: Zebra, ospfd and ospfd6. These daemons perform the following functions:
- Create OSPF IPv4 router
- Create OSPF IPv6 router

The configuration is described in Annex A.2.10.2.

- CAFE_F: runs following daemons: Zebra, bgpd, ospfd and ospfd6. These daemons perform the following functions:
- Create BGP IPv4 router for the CAFE_F
- Create BGP IPv6 router for the CAFE_F
- Create the BGP peer with AS 62 (WUT)
- Create OSPF IPv4 router
- Create OSPF IPv6 router

The configuration is described in Annex A.2.10.3.

- CAFE_61: runs following daemons: Zebra, ospfd and ospfd6. This daemons perform following functions:
    - Create OSPF IPv4 router
    - Create OSPF IPv6 router

The configuration is described in Annex A.2.10.4.

AS62:

- CAFE_A: runs following daemons: Zebra, bgpd, ospfd and ospfd6. This daemons perform following functions:
    - Create BGP IPv4 router for the CAFE_A
    - Create BGP IPv6 router for the CAFE_A
    - Create the BGP peer with AS 51 (TID)
    - Create the BGP peer with AS 63 (WUT)
    - Create OSPF IPv4 router
    - Create OSPF IPv6 router

The configuration is described in Annex A.2.10.5.

- CAFE_B: runs following daemons: Zebra, ospfd and ospfd6. This daemons perform following functions:
  - Create OSPF IPv4 router
  - Create OSPF IPv6 router.

The configuration is described in Annex A.2.10.6.

- CAFE_C: runs following daemons: Zebra, bgpd, ospfd and ospfd6. This daemons perform following functions:
  - Create BGP IPv4 router for the CAFE_C
  - Create BGP IPv6 router for the CAFE_C
  - Create the BGP peer with AS 62 (WUT)
  - Create OSPF IPv4 router
  - Create OSPF IPv6 router

The configuration is described in Annex A.2.10.7.

- CAFE_62: runs following daemons: Zebra, ospfd and ospfd6. This daemons perform following functions:
- Create OSPF IPv4 router
- Create OSPF IPv6 router

The configuration is described in Annex A.2.10.8.

AS63:

- CAFE_G: runs following daemons: Zebra, bgpd, ospfd and ospfd6. This daemons perform following functions:
- Create BGP IPv4 router for the CAFE_G
- Create BGP IPv6 router for the CAFE_G
- Create the BGP peer with AS 61 (WUT)
- Create the BGP peer with AS 62 (WUT)

The configuration is described in Annex A.2.10.9.

3.5.3 Forwarding Keys to neighbouring border CAFEs

The concept of these forwarding keys does not differ from the general definition for a forwarding key. The only difference is that these keys will instruct a border CAFE about what interface to choose when forwarding outgoing traffic to a neighbouring testbed.

3.5.3.1 Forwarding Keys to neighbouring border CAFEs in TID

In the case of TID only two keys of this type are needed.

- Key for forwarding traffic from CAFE-TIDPT to CAFE_PTID (AS 41), which was assigned the value 0xfc
- Key for forwarding traffic from CAFE-TIDWUT to CAFE_A (AS 62), which was assigned the value 0xfd

The complete definition for the forwarding keys can be found in Annex A.1.2.

3.5.3.2 Forwarding Keys to neighbouring border CAFEs in PTL[PTL]

The PTL forwarding keys of this type were:

- Key 0xc1 for forwarding traffic from CAFE_PTLTID to CAFE-TIDPT (AS 51).
- Key 0xb1 for forwarding traffic from CAFE-PTLWUT to CAFE_D (AS 61).

The complete definition for the forwarding keys can be found in Annex A.3.2.

3.5.3.3 Forwarding Keys to neighbouring border CAFE in WUT

In the case of WUT local testbed all its AS are connected with each other, moreover AS61 is connected with PTL local testbed (AS41) and AS62 is connected with TID local testbed (AS51). Furthermore, on each AS interconnection all CoS are supported. As a consequence following forwarding keys are configured:

In AS 61:

- key for forwarding traffic from CAFE_D to CAFE-PTWUT (in AS 41), which was assigned the value 0x41 for PR traffic and 0x42 for BTBE traffic,
- key for forwarding traffic from CAFE_D to CAFE_G (AS 63), which was assigned the value 0x63for PR traffic and 0x93 for BTBE traffic,
- key for forwarding traffic from CAFE_F to CAFE_C (AS 62), which was assigned the value 0x62for PR traffic and 0x92 for BTBE traffic,

In AS 62:

- key for forwarding traffic from CAFE_A to CAFE-TIDWUT (in AS 51), which was assigned the value 0x51 for PR traffic and 0x52 for BTBE traffic,
- key for forwarding traffic from CAFE_C to CAFE_G (AS 63), which was assigned the value 0x63for PR traffic and 0x93 for BTBE traffic,
- key for forwarding traffic from CAFE_C to CAFE_F (AS 62), which was assigned the value 0x61for PR traffic and 0x91 for BTBE traffic,

In AS 63:

- key for forwarding traffic from CAFE_G to CAFE_D (in AS 61), which was assigned the value 0x61for PR traffic and 0x91 for BTBE traffic,
- key for forwarding traffic from CAFE-G to CAFE_A (AS 62), which was assigned the value 0x62for PR traffic and 0x92 for BTBE traffic.

The complete definition for the forwarding keys can be found in Annex A.

3.5.4 Paths for outgoing traffic

Once a local RAE has been fully configured for characterizing paths inside a local AS, it needs to advertise the characteristics of the links to neighboring AS, but only for outgoing traffic, in other words, traffic with source in the CSs deployed inside the local AS. The parameters that have to be specified are the following:

- AS Sink.
- CoS for the outgoing traffic.
- BW.
- IPLR.
- IPTD.

3.5.4.1 Paths for outgoing traffic for TID

Two outgoing paths have to be defined for TID testbed:

- One going to AS 62 (WUT).
- The Other going to AS 41 (PTL).

These two paths will be assigned different QoS Parameters according the CoS, leading to four different paths with the following information:

AS Sink	COS	BW	IPLR	IPTD
62	BtBE	10000000	0.000005	0.005
62	PR	20000000	0.000001	0.001
41	BtBE	10000000	0.000008	0.010
41	PR	16000000	0.000002	0.002

Table 14: Outgoing paths for TID as defined in RAE.

A more complete description can be found at 3.2.4.

3.5.4.2 Paths for outgoing traffic in PTL

Two outgoing paths are defined for PTL testbed:

- One going to AS 51 (TID)
- The other going to AS 61 (WUT)

These two paths will be assigned different QoS Parameters according the CoS, leading to four different paths with the following information.

AS Sink	COS	BW	IPLR	IPTD
51	BtBE	10000000	0.001	0.1
51	PR	20000000	0.0001	0.01
61	BtBE	10000000	0.000001	0. 1
61	PR	20000000	0.00001	0.01

Table 15: Outgoing paths for PTL as defined in RAE.

The complete description is found at 3.4.4.

3.5.4.3 Paths for outgoing traffic in WUT

Two outgoing paths are defined in WUT local testbed:

- Internal paths between domains in WUT:
  - path from AS 61 to AS 62,
  - path from AS 62 to AS 61,
  - path from AS 61 to AS 63,
  - path from AS 63 to AS 61,
  - path from AS 62 to AS 63,
  - path from AS 63 to AS 61,
- External paths outgoing from WUT:
  - path from AS 61 to AS 41 (located in PTL),
  - path from AS 62 to AS 51 (located in TID).

Above paths will be assigned different QoS Parameters according the CoS, see Table 16 for details.

AS Sink	COS	BW [MB]	IPLR	IPTD [s]
AS 61				
62	BtBE	100	0.000001	0.001
62	PR	100	0.000001	0.0005
63	BtBE	100	0.000001	0.001
63	PR	100	0.000001	0.0005
41	BtBE	100	0.000001	0.001
41	PR	100	0.000001	0.0005
AS 62				
61	BtBE	100	0.000001	0.001
61	PR	100	0.000001	0.0005
63	BtBE	100	0.000001	0.001
63	PR	100	0.000001	0.0005
51	BtBE	100	0.000001	0.001
51	PR	100	0.000001	0.0005
AS 63				
61	BtBE	100	0.000001	0.001
61	PR	100	0.000001	0.0005
63	BtBE	100	0.000001	0.001
63	PR	100	0.000001	0.0005

Table 16: Outgoing paths for WUT as defined in RAE.

3.5.5 Peering AS/CAFE Mappings

A Peering AS/CAFE Mapping is used to define which border CAFEs manage the ends of link connecting two neighboring AS. The information is needed to define this sort of mapping is the following:

- AS linked.
- CoS.
- Source CAFE.
- Sink CAFE.
- CME in charge of serving the neighbouring AS.
- Path capacity.

3.5.5.1 Peering AS/CAFE Mappings for TID

In the case the following Peering have been defined:

- AS 51 to AS 41, with CAFE-TIDPT as source and CAFE_PTID as sink, using the CME at AS 41.
- AS 41 to AS 51, with CAFE-PTTID as source and CAFE_PTID as sink, using the CME at AS 41.
- AS 51 to AS 62 with CAFE-TIDWUT as source and CAFE_A as sink, using the CME at AS 62.
- AS 62 to AS 41, with CAFE_A as source and CAFE_TIDWUT as sink, using the CME at AS 62.

These peerings have to be defined both for BTBE an PR, the difference being the path capacity:

- 100M for Pr.
- 50M for BTBE.

The complete definition can be found at A.1.5.

3.5.5.2 Peering AS/CAFE Mappings in PTL

The following peering mappings have been defined:

- AS 41 to AS 51, with CAFE-PTLTID as source and CAFE_PTID as sink, using the CME at AS 51
- AS 41 to AS 61, with CAFE-PTLWUT as source and CAFE_D as sink, using the CME at AS 61
- AS 51 to AS 41, with CAFE_PTID as source and CAFE-PTLTID as sink, using the CME at AS 51
- AS 61 to AS 41, with CAFE_D as source and CAFE-PTLWUT as sink, using the CME at AS 61

These peering have been defined both for BTBE and PR, the difference being the Path Capacity:

- 100M for Pr
- 50M for BTBE

The complete definition can be found at A.3.4.

3.5.5.3 Peering AS/CAFE Mappings in WUT

In the case of WUT domains, the following peering AS/CAFE mappings are configured:

For AS 61:

- CAFE_D (AS 61) to CAFE_PTWUT (AS 41) with key 0x41 for PR and 0x42 for BTBE.
- CAFE_F (AS 61) to CAFE_C (AS 62) with key 0x62 for PR and 0x92 for BTBE.
- CAFE_D (AS 61) to CAFE_G (AS 63) with key 0x63 for PR and 0x93 for BTBE.

For AS 62:

- CAFE_A (AS 62) to CAFE_TIDWUT (AS 51) with key 0x51 for PR and 0x52 for BTBE.
- CAFE_C (AS 62) to CAFE_F (AS 61) with key 0x61 for PR and 0x91 for BTBE.

- CAFE_A (AS 6s) to CAFE_G (AS 63) with key 0x63 for PR and 0x93 for BTBE.

For AS 62:

- CAFE_G (AS 63) to CAFE_D (AS 61) with key 0x61 for PR and 0x91 for BTBE..
- CAFE_G (AS 63) to CAFE_A (AS 62) with key 0x62 for PR and 0x92 for BTBE.

Above peering are defined both for BTBE an PR with Path Capacity equal to 100 Mb for each class (BTBE, PR).

3.5.6 Peering CAFEs/Keys Mappings

These CAFE/Keys mappings do not conceptually differ from those created inside a testbed. The only difference is that they associated border CAFEs in different AS. As those previous mappings, the following parameters have to be specified. Traffic as always is considered going from server to client

- Source CAFE.
- Sink CAFE.
- Forwarding key at the source CAFE.
- Link capacity.
- CoS.

3.5.6.1 Peering CAFEs/Keys Mappings in TID

In the case of TID the following peering have been defined:

- CAFE-TIDWUT (AS 51) to CAFE_A (AS 62) with key 0xfc.
- CAFE_TIDPT (AS 51) to CAFE_PTTID (AS 41) with key 0xfc.

This Peering have been configured for both CoS, the only difference being the path capacity

- For PR CoS, capacity is assigned 100MB.
- For BtBE CoS capacity is assigned 50MB.

Full configuration is described in Annex A.1.6

3.5.6.2 Peering CAFEs/Keys Mappings in PTL

The following peering mappings have been defined:

- CAFE-PTLTID (AS41) to CAFE_PTID (AS51) with key 0xc1 for PR and 0xca for BTBE
- CAFE-PTLWUT (AS41)to CAFE_D (AS61) with key 0xb1 for PR and 0xba for BTBE

Path Capacity assigned for each CoS:

- 100M for Pr
- 50M for BTBE

The complete definition can be found at A.3.5

3.5.6.3 Peering CAFEs/Keys Mappings in WUT

In the case of WUT domains, the following peering AS/CAFE mappings are configured:

For AS 61:

- CAFE_D (AS 61) to CAFE_PTWUT (AS 41) with key 0x41 for PR and 0x42 for BTBE.
- CAFE_F (AS 61) to CAFE_C (AS 62) with key 0x62 for PR and 0x92 for BTBE.
- CAFE_D (AS 61) to CAFE_G (AS 63) with key 0x63 for PR and 0x93 for BTBE.

For AS 62:

- CAFE_A (AS 62) to CAFE_TIDWUT (AS 51) with key 0x51 for PR and 0x52 for BTBE.

- CAFE_C (AS 62) to CAFE_F (AS 61) with key 0x61 for PR and 0x91 for BTBE.
- CAFE_A (AS 6s) to CAFE_G (AS 63) with key 0x63 for PR and 0x93 for BTBE.

For AS 62:

- CAFE_G (AS 63) to CAFE_D (AS 61) with key 0x61 for PR and 0x91 for BTBE.
- CAFE_G (AS 63) to CAFE_A (AS 62) with key 0x62 for PR and 0x92 for BTBE.

Above peering are defined both for BTBE an PR with Path Capacity equal to 100 Mb for each class (BTBE, PR).

Full configuration is described in Annex A.2.5

3.5.7 ROOT CRE configuration

Root CRE was deployed in PTL testbed accessible throughout the federated testbed. The naming authorities and respective authoritative CRE IPs are as shown in the below table.

Naming authority	CRE authoritative	
	IPv4	IPv6
tid1.es	10.95.51.2	2a02:9008:0:1911:0:50:56a3:4e
wut2.pl	10.95.51.2	2a02:9008:0:1911:0:50:56a3:4e
ptl3.cy	10.50.50.5	3105::5

Table 17: CRE root configuration in PTL.

3.6 Remote Access

TID has deployed a Remote Access to allow remote clients to connect to the Federated Testbed, a special requirement for this access is that it must support IPv6 and IPv4connectivity. The remote access should provide the clients with:

- IPv4 address.
- IPv6 address.
- Routing to reach Federated Testbed.

To deploy the remote access, TID has setup an OpenVPN alpha v. 2.3 (the alpha version have ipv6 native support) working as a bridged VPN tunnel over UDP protocol.

TID has deployed a gateway with a public IP address (130.206.214.70) and a UDP port (2012) to allow clients to reach the TID Remote Access server from the Internet. This gateway performs a NAT (Network Address Translation) to reach the internal IP address of the server (10.95.51.66).

TID has also created several credentials to allow clients to connect to the server. These credentials are based on client authentication methods based on certificates.

TID has set up the OpenVPN Server to assign IPv4 address dynamically to the clients. The routing to reach the networks is forwarded using OpenVPN Server. Quagga has been configured to assign IPv6 addresses, and propagate different routes.

The OpenVPN configuration file is attached in Annex A.1.9.3.

The server has been installed in the machine “CAFE-TID2”

3.7 Use Cases on the Federated Testbed

This section is allocated for the description of how the use cases have been transformed in a real demo that can be shown and tested over the federated testbed.

Note that since the basic entry point for demonstration will be the external access deployed at TID, the description is done from TID's point of view.

3.7.1 Use Case 1: Adaptable and efficient content distribution

This Use Case demonstrates how COMET can redirect content queries from CCs in TID to different CSs deployed throughout the Federated Testbed according to CCs and CSs CoS, by using a single Content Name for all the server population.

It will be also demonstrated how in case of multiple available CSs, the server selection will be governed by the server load, while for multiple available paths the criterion will be the match between the path QoS characteristics and the CSs QoS requirements.

3.7.1.1 Prerequisites

As depicted in the figure, the Federated Testbed has been configured as follows

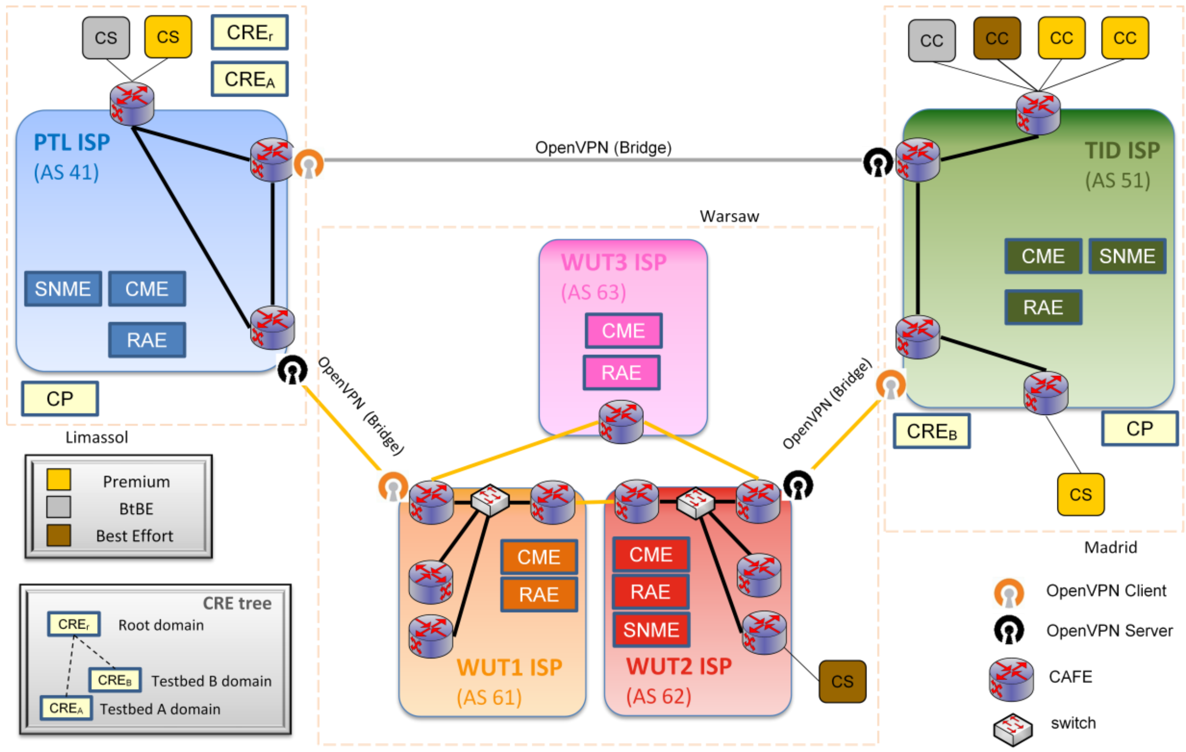


Figure 5: Testbed Configuration for Use Case 1

- CCs for all CoS will be deployed in TID (for demo purposes this CCs will join the testbed by using the Remote Access)
- A Pr Streaming CS will be deployed in TID, which will be later substituted by a Pr VoD CS at the same location.
- In PTL there will be both one Pr Streaming CS and one BtBE.
- In WUT a BE CS will be also deployed.

Since the testbed is polarised, CCs at TID, servers at other locations, the testbed has been configured in order to ensure that BtBE CSs at PTL are accessed through the direct link between

PTL and TID, while PR CSs are diverted through WUT ISPs. To achieve this aims the following configuration actions have been taken.

In PTL, the RAE has been configured to make the PTL-TID link less attractive for PR Users than the PTL-WUT. The actual used configuration is as follows:

Link	IPLR	IPTD	BW
PTL-TID	1e-4	0.010	20e6
PTL-WUT	1e-6	0.001	20e6

Table 18: QoS Parameters for TID and WUT links as seen from PTL

For testing purposes a content name called tid1.es/demo_b has been created, comprising the following individual Content Sources containing:

- One for the Pr CS at TID.
- One for the Pr CS at PTL.
  - This Content Source is of lower priority, so that this is only taken into account when the previous Content Source is inaccessible.
- One for the BtBE CS at PTL.
- One for the BE CS at WUT.

The difference between the PR and BtBE Content Sources at PTL is that the first one has QoS targets greater than the QoS definition in the PTL-TID direct link, so this link will not be the first choice for the decision process and the indirect path via WUT will be selected instead. The values set for this Content Source are:

- IPTD 0.001.
- IPLR 0.4.

The use case, as explained in the following section, will require the removal of the streaming servers and their substitution for VoD. In order to avoid reinserting the information for each test, an auxiliary Content Name tid1.es/demo_c, has been created for ease of use containing the VoD server at TID only.

As explained in section 3.2.9, the decision algorithm in TID has been configured to:

- Be strict on Load Server, to avoid CSs on High occupation status being eligible.
- Be strict on IPTD, to avoid paths which do not match the QoS requirements defined in the Content Sources and make this parameter the most important for the decision algorithm.

3.7.1.2 Description

The storyline for this case is as follows:

- A content provider is distributing premium video inside a ISP
- Because of users' high demand, its servers become overloaded.
- Hosting costs are unacceptable in its home domain so it takes the decision of shifting part of their operation to external domains.
- The success of this relocation enables him to distribute video for lower CoSs (BTBE and BE) from those external domains.

Next Figure show the CS and CC population used for illustrating this case, as explained in the previous sections.

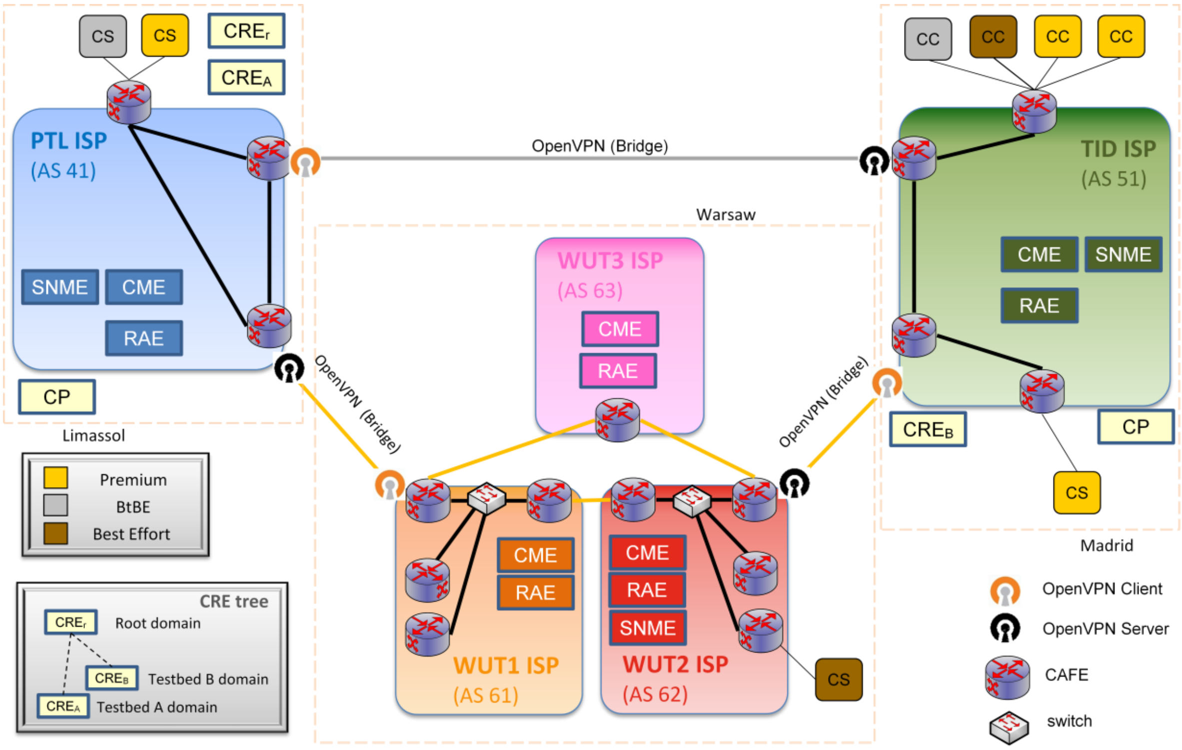


Figure 6: Testbed configuration for Use Case 1

As it can be seen in the figure, CCs for all CoS have been deployed in TID testbed, while in terms of CSs, TID will have a Pr one, PTL both a Pr and a BTBE and WUT a BE. Paths inside the testbed have been configured to force Pr traffic via WUT and BTBE through the direct link.

The first stage simply shows how in a normal case, requesting tid1.es/demo_b from a PR CC at TID launches a content retrieval from the local Pr CSs deployed at TID, as illustrated in the next figure. The CME will detect that two content sources have been defined in this content name for CoS and as the local content source has been assigned a higher priority than the remote one, the local one will be evaluated first and chosen for retrieval. CS's edge CAFE will be configured to intercept the traffic with source in that CS and forward it to the CAFEs in the path until the requesting CC.

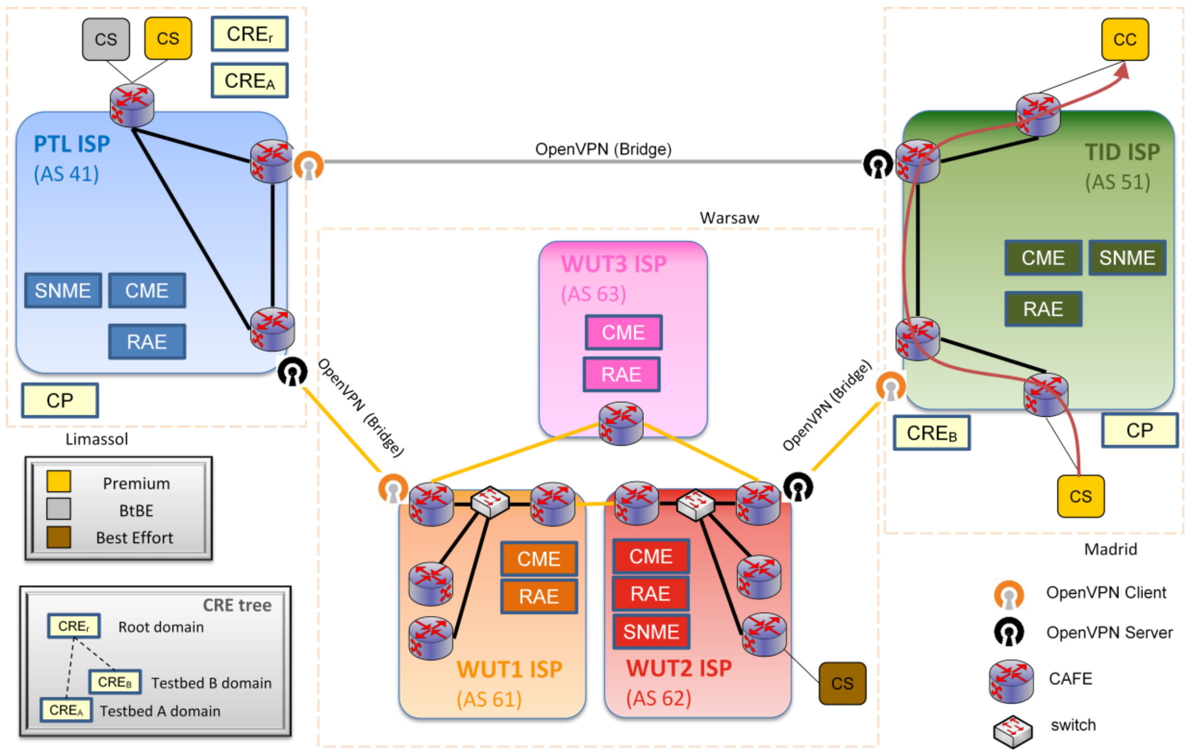


Figure 7: Normal Retrieval for PR in-testbed

The next case is more interesting for COMET purposes. Now impairments are introduced in the Pr CS TID, making it non-eligible for content retrieval. When tid1.es/demo_b is requested from the same Pr CC at TID, TID CME falls back on the other content source, the one with lower priority, located at PTL. The Pr CS at PTL is in a low occupation status, so it will become a fitting candidate for content retrieval.

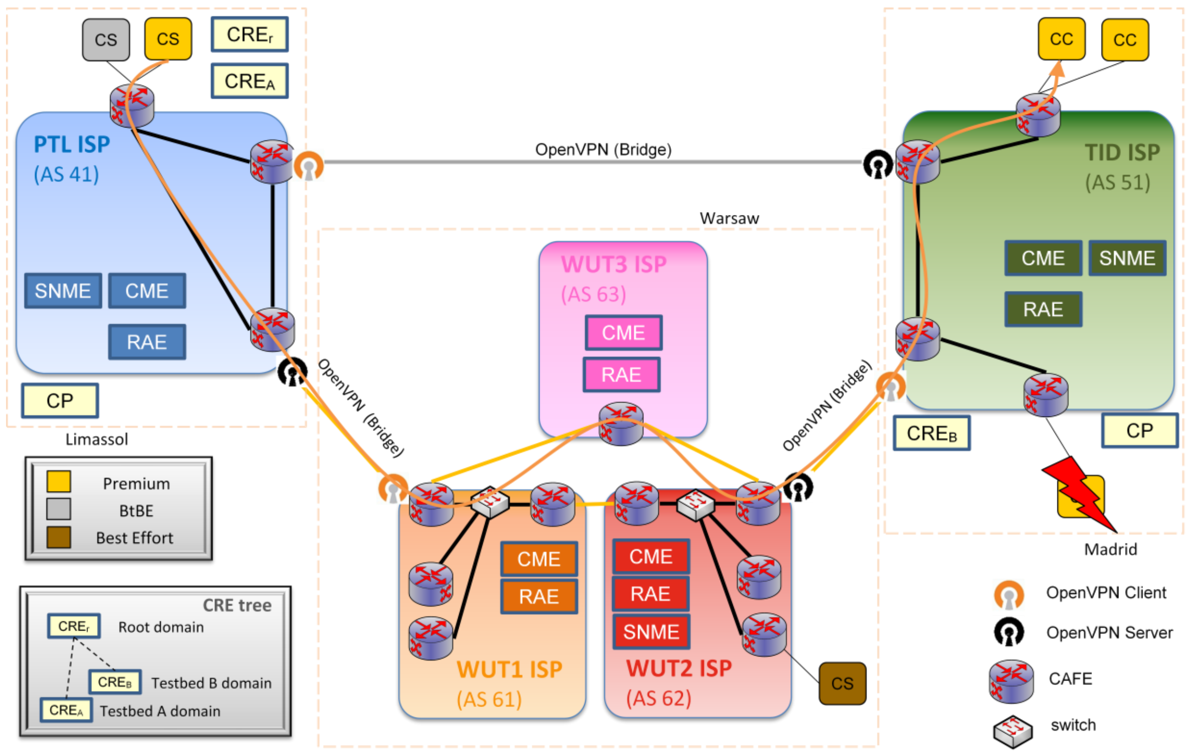


Figure 8: Remote PR CS chosen when local PR CS is overloaded

According to the path configuration in PTL RAEs and the decision algorithm tuning in TID CME, the path selected will be the one crossing WUT testbed, as illustrated in the following figure. PTL CME will configure the CS's edge CAFE configured to intercept the traffic with source in that CS and forward it to the CAFEs in the path until the requesting CC at TID.

Next step will involve the deployment of a BtBE CC at TID and a BtBE CS at PTL. In this case, when the same content name, tid1.es/demo_b is requested, the retrieval is directed to the BtBE CS at PTL. The path selected is the direct link PTL-TID, according to the path configuration in PTL RAEs and the decision algorithm tuning in TID CME.PTL CME will configure the CS's edge CAFE configured to intercept the traffic with source in that CS and forward it to the CAFEs in the path until the requesting CC at TID.

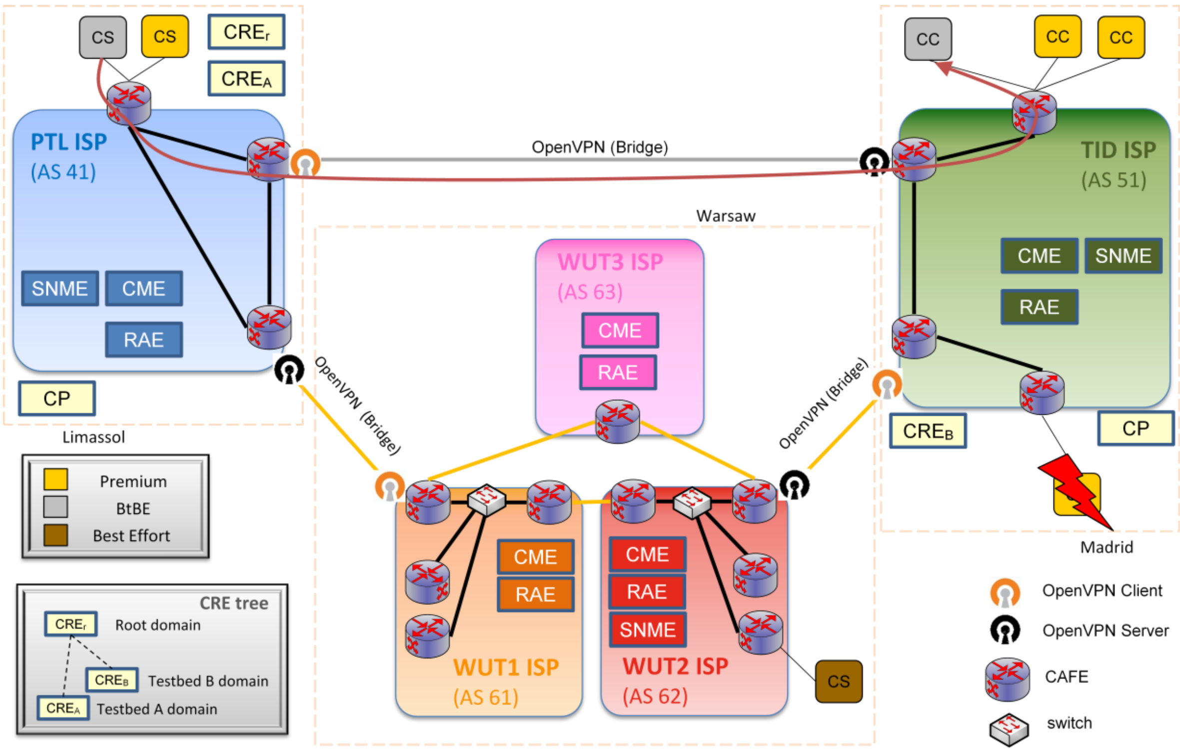


Figure 9: Different Path selection for BtBE

The last example shows how if a BE CC is deployed at TID and a BE CS at WUT, COMET is able to map the CC CoS with the right content source, and start retrieval from the suitable CS at WUT. No CAFE configuration takes place now and the traffic is routed according to the default IP routing map.

Finally, the streaming servers are removed (as though the live broadcast of an event had finished), impairments are removed from TID servers and a VoD CS is deployed at TID under the umbrella of the same content name.

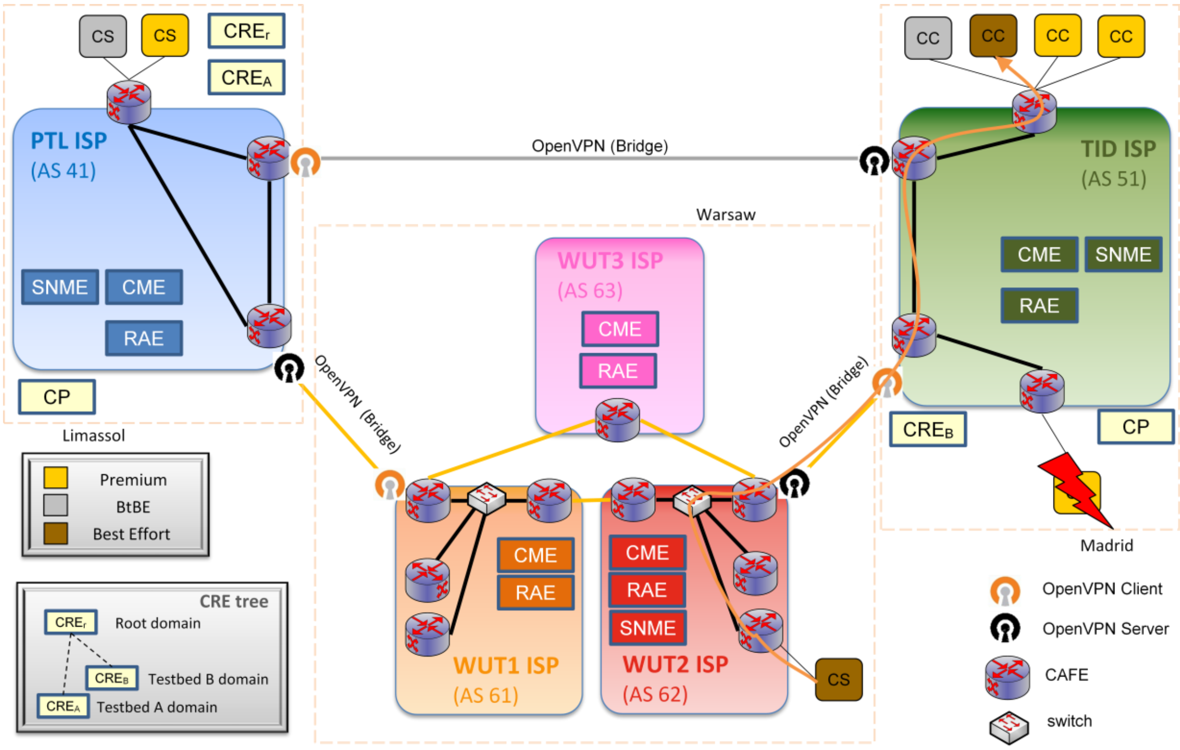


Figure 10: Different Server Selection for BE

When a Pr CC request tid1.es/demo_b, the content will be downloaded from the freshly deployed VoD CS. TID CME will configure the CS's edge CAFE configured to intercept the traffic with source in that CS and forward it to the CAFEs in the path until the requesting CC at TID.

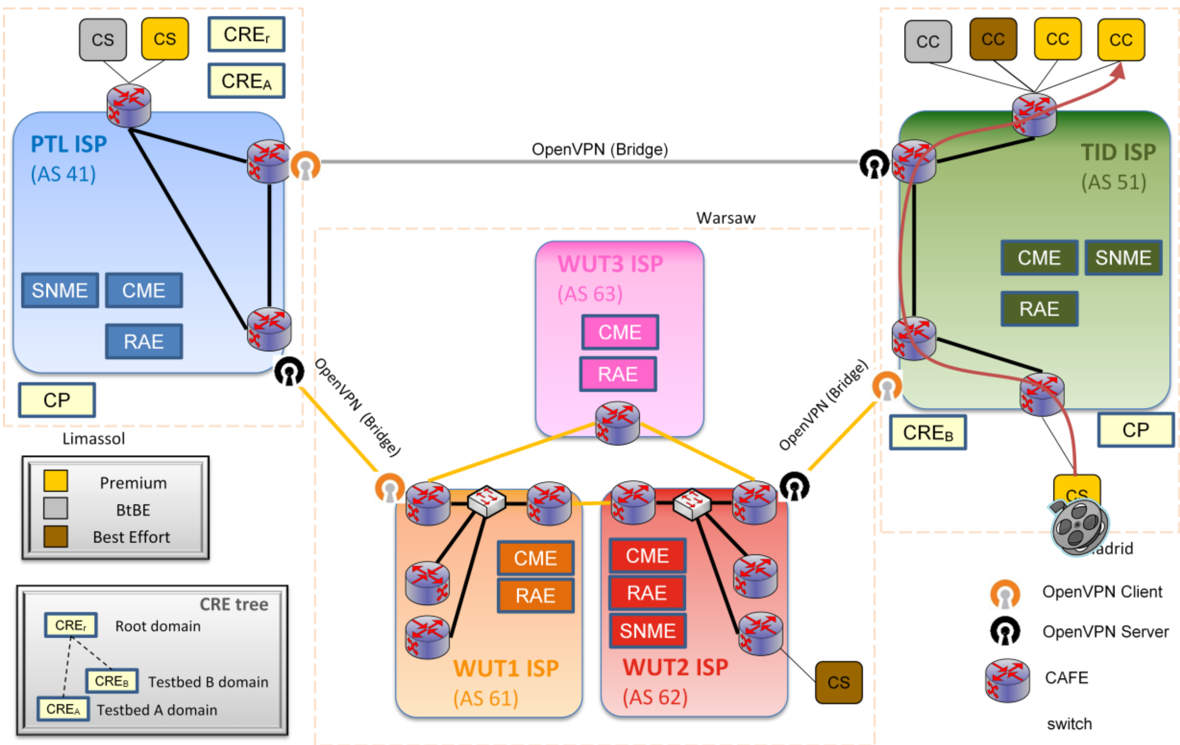


Figure 11: Server Selection when streaming CSs are replaced by VoD ones

3.7.2 Use Case 4: Offloading

This Use Case seeks to demonstrate how COMET can divert Pr Users from PR streaming Server to P2P overlays, in case the Pr services are overloaded. Denial of Service will be thus avoided in any circumstances and the service always provided, although quality will be traded off in the exchange.

3.7.2.1 Prerequisites

The only requirement for this case is a Content Name, tid1.es/demo_p2p with two Content Sources:

- One for the Pr Streaming CS at TID.
- One for the BE P2P CS at TID, with a lower priority than the Pr, ensuring that the first choice assessed by CME will be always the Pr one and that P2P will only be taking into account if no Pr CS is available.
- In order to make the use case more complete, more P2P CS can be deployed in other ISPs, for instance, at WUT.

3.7.2.2 Description

The following image depicts how the Federated Testbed is prepared to demonstrate Use Case 4:

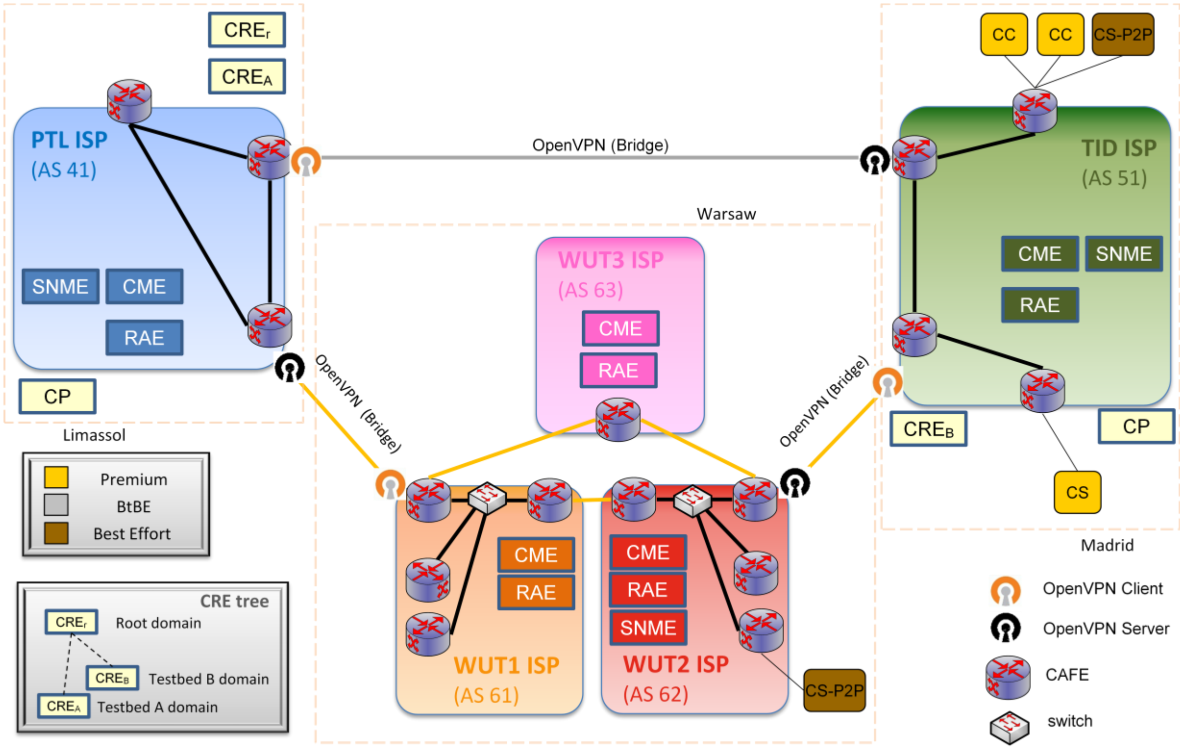


Figure 12: Use Case 4: Initial Situation

Basically, Pr CSs will be deployed at TID, as well as a Pr CS and two CS-P2P, all under the umbrella of a single content name tid1.es/demo_P2P.

The first step is the same as the Use Case 1. A Pr CC will request tid1.es/demo_P2P and the content will be retrieved from the Pr CS, with the same assumptions that in the precedent case.

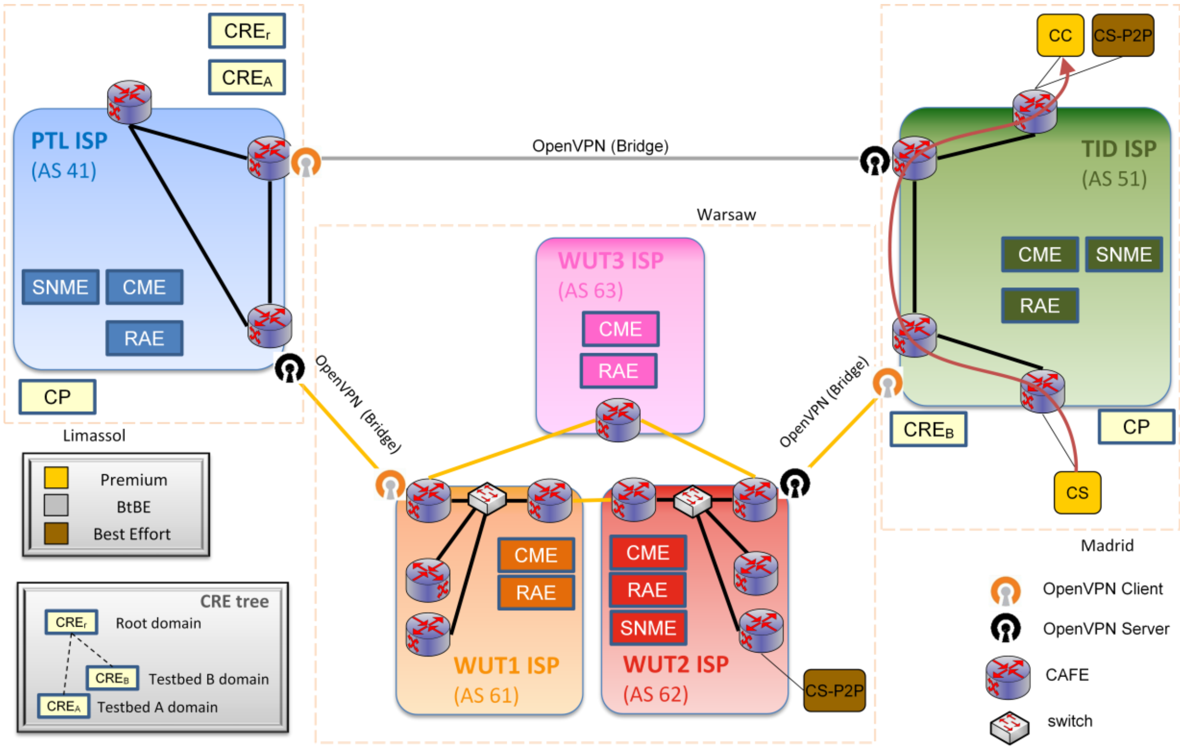


Figure 13: Use Case 4: Normal Retrieval

After that, impairments are introduced in the Pr CSs, leading up to a second Pr CC being redirected to the P2P server at TID. Since the P2P services are BE by default, no CAFE configuration is performed in this case.

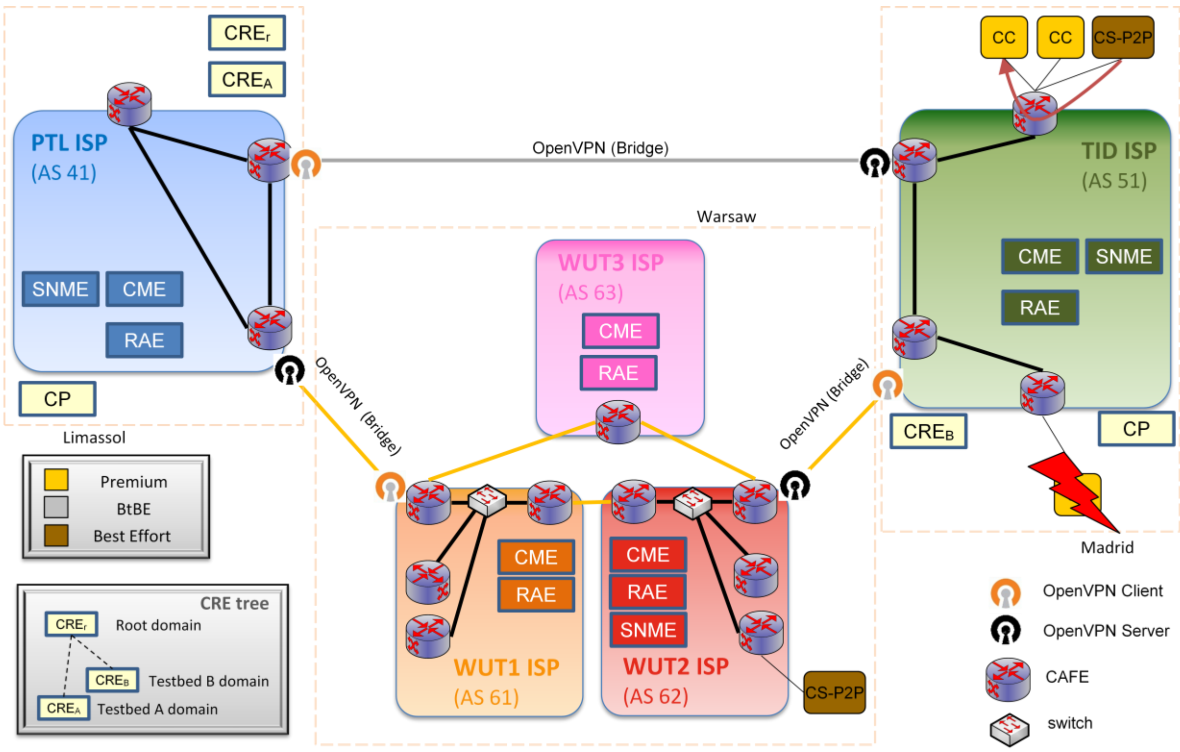


Figure 14: Use Case 4: P2P offloading

As explained before, this case can be complemented by deploying the CS-P2P in other locations of the federated testbed, demonstrating that location is not a determining factor in the decision process for this case, since it is the priority mechanism that takes over according to the load of the Pr Servers.

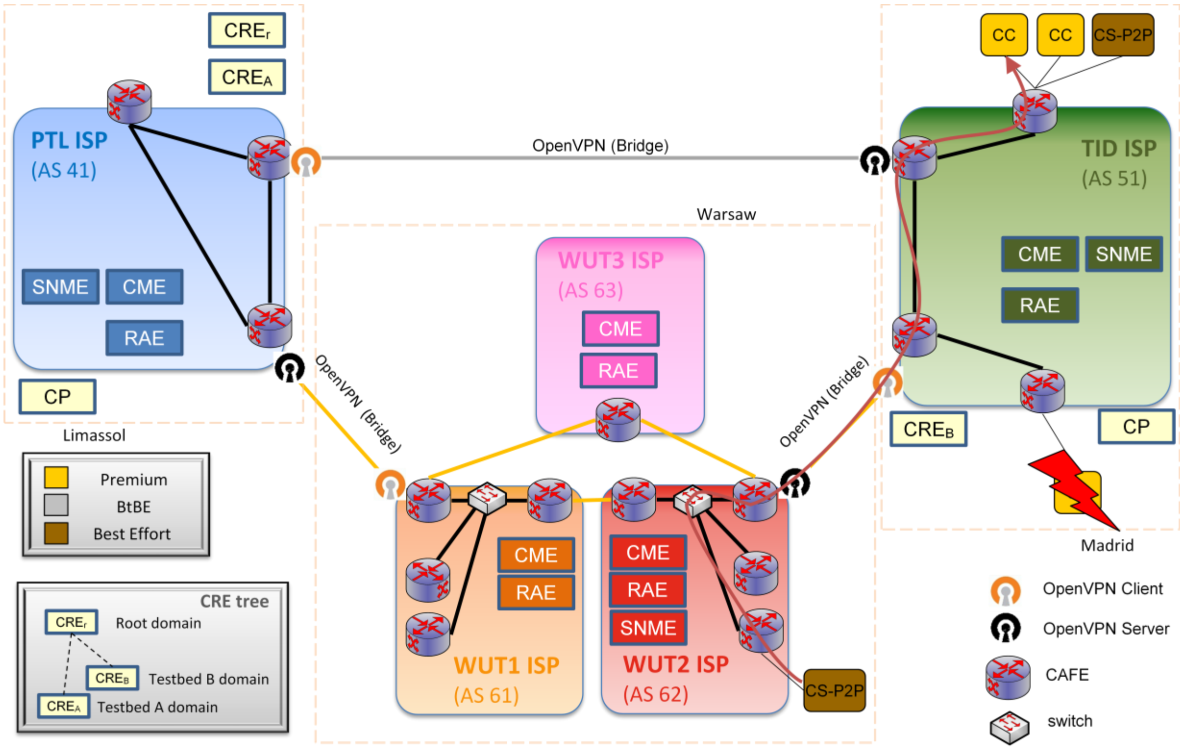


Figure 15: Use Case 4: P2P Offloading to external services

Section 4.1, allocated to functional testing, describes more in detail how COMET behaves in each stage of the uses cases.

4 Testing

4.1 Functional Tests

4.1.1 Adaptable and efficient content distribution (Use Case 1)

4.1.1.1 FUN-UC1-001

This test demonstrates how RAEs in different ISPs gather information about the paths connecting them (Routing Awareness Function)

In the federated testbed there are five ASes, as explained in section 3: 41 (ISP PTL), 51 (ISP TID), 61 (ISP WUT1), 62 (ISP WUT2) and 63 (ISP WUT3). From the point of view of the RAE RAE-TID [2a02:9008:0:1911:0:50:56a3:51], initially this entity only stores local paths and those linking to neighbouring ISP. These paths are characterized with the following parameters: path destination, server network, CoS (with the different possible paths), AS to cross through and QoS parameters(as BW, packet delay and packet loss).

The final situation, after allowing the convergence path is that the CRE at TID stores path to reach every AS in the testbeb with their associated QoS information. By Executing in RAE-TID the command sqlite –header db51.sqlite3 “select * from paths;” all the possible paths and their associated information is retrieved, as shown in the following listing.

```
prefix|classid|edge|path|delay|loss|bw
2a02:9008:0:1913::/64|1|#62|51|0.000500000023748726|9.99999997475243e-07|20000000.0
2a02:9008:0:1913::/64|1|#41|51|0.000500000023748726|9.99999997475243e-07|20000000.0
2a02:9008:0:1913::/64|2|#62|51|0.0010000000474974 5|9.99999997475243e-07|20000000.0
2a02:9008:0:1913::/64|2|#41|51|0.00100000004749745|9.99999997475243e-07|20000000.0
2a02:9008:0:1912::/64|1|#62|51|0.000500000023748726|9.99999997475243e-07|20000000.0
2a02:9008:0:1912::/64|1|#41|51|0.000500000023748726|9.99999997475243e-07|20000000.0
2a02:9008:0:1912::/64|2|#62|51|0.00100000004749745|9.99999997475243e-07|20000000.0
2a02:9008:0:1912::/64|2|#41|51|0.00100000004749745|9.99999997475243e-07|20000000.0
10.203.62.0/27|2|#62|62 51|0.0020000000949949|1.99999908545578e-06|20000000.0
10.203.62.0/27|2|#62|62 61 41 51|0.00800000037997961|7.99997269496089e-06|16000000.0
10.203.62.0/27|2|#62|62 63 61 41 51|0.0120000010356307|1.19999367598211e-05|16000000.0
10.203.62.0/27|2|#41|62 51|0.00300000002607703|2.99999715025479e-06|20000000.0
10.203.62.0/27|2|#41|62 61 41 51|0.00700000021606684|6.9999800871301e-06|16000000.0
10.203.62.0/27|2|#41|62 63 61 41 51|0.0110000008717179|1.09999473352218e-05|16000000.0
2001:67c:24cc:31c2::/64|2|#62|61 41 51|0.00600000005215406|5.99998611505725e-06|16000000.0
2001:67c:24cc:31c2::/64|2|#62|61 62 51|0.00400000018998981|3.99999407818541e-06|20000000.0
2001:67c:24cc:31c2::/64|2|#62|61 63 62 51|0.00700000021606684|6.9999800871301e-06|16000000.0
2001:67c:24cc:31c2::/64|2|#41|61 41 51|0.00499999988824129|4.99999123348971e-06|16000000.0
2001:67c:24cc:31c2::/64|2|#41|61 62 51|0.00500000035390258|4.99998986924766e-06|20000000.0
2001:67c:24cc:31c2::/64|2|#41|61 63 62 51|0.00800000037997961|7.99997269496089e-06|16000000.0
2001:67c:24cc:31c2::/64|1|#62|61 41 51|0.0309999994933605|2.999968273798e-05|10000000.0
2001:67c:24cc:31c2::/64|1|#62|61 62 51|0.0109999999403954|1.19999540402205e-05|10000000.0
2001:67c:24cc:31c2::/64|1|#62|61 63 62 51|0.0214999988675117|2.09998379432363e-05|10000000.0
2001:67c:24cc:31c2::/64|1|#41|61 41 51|0.0305000003427267|2.89997115032747e-05|10000000.0
2001:67c:24cc:31c2::/64|1|#41|61 62 51|0.0115000000223517|1.29999416458304e-05|10000000.0
2001:67c:24cc:31c2::/64|1|#41|61 63 62 51|0.0219999980181456|2.19998164538993e-05|10000000.0
10.203.61.240/30|1|#62|61 41 51|0.0309999994933605|2.999968273798e-05|10000000.0
10.203.61.240/30|1|#62|61 62 51|0.0109999999403954|1.19999540402205e-05|10000000.0
10.203.61.240/30|1|#62|61 63 62 51|0.0214999988675117|2.09998379432363e-05|10000000.0
10.203.61.240/30|1|#41|61 41 51|0.0305000003427267|2.89997115032747e-05|10000000.0
10.203.61.240/30|1|#41|61 62 51|0.0115000000223517|1.29999416458304e-05|10000000.0
10.203.61.240/30|1|#41|61 63 62 51|0.0219999980181456|2.19998164538993e-05|10000000.0
10.203.61.240/30|2|#62|61 41 51|0.00600000005215406|5.99998611505725e-06|16000000.0
10.203.61.240/30|2|#62|61 62 51|0.00400000018998981|3.99999407818541e-06|20000000.0
10.203.61.240/30|2|#62|61 63 62 51|0.00700000021606684|6.9999800871301e-06|16000000.0
10.203.61.240/30|2|#41|61 41 51|0.00499999988824129|4.99999123348971e-06|16000000.0
10.203.61.240/30|2|#41|61 62 51|0.00500000035390258|4.99998986924766e-06|20000000.0
```

```
10.203.61.240/30|2|#41|61 63 62 51|0.00800000037997961|7.99997269496089e-06|16000000.0
10.203.61.0/27|1|#62|61 41 51|0.0309999994933605|2.999968273798e-05|10000000.0
10.203.61.0/27|1|#62|61 62 51|0.0109999999403954|1.19999540402205e-05|10000000.0
10.203.61.0/27|1|#62|61 63 62 51|0.0214999988675117|2.09998379432363e-05|10000000.0
10.203.61.0/27|1|#41|61 41 51|0.0305000003427267|2.89997115032747e-05|10000000.0
10.203.61.0/27|1|#41|61 62 51|0.0115000000223517|1.29999416458304e-05|10000000.0
10.203.61.0/27|1|#41|61 63 62 51|0.0219999980181456|2.19998164538993e-05|10000000.0
10.203.61.0/27|2|#62|61 41 51|0.00600000005215406|5.99998611505725e-06|16000000.0
10.203.61.0/27|2|#62|61 62 51|0.00400000018998981|3.99999407818541e-06|20000000.0
10.203.61.0/27|2|#62|61 63 62 51|0.00700000021606684|6.9999800871301e-06|16000000.0
10.203.61.0/27|2|#41|61 41 51|0.00499999988824129|4.99999123348971e-06|16000000.0
10.203.61.0/27|2|#41|61 62 51|0.00500000035390258|4.99998986924766e-06|20000000.0
10.203.61.0/27|2|#41|61 63 62 51|0.00800000037997961|7.99997269496089e-06|16000000.0
2001:67c:24cc:31c3::/64|1|#62|61 41 51|0.0309999994933605|2.999968273798e-05|10000000.0
2001:67c:24cc:31c3::/64|1|#62|61 62 51|0.0109999999403954|1.19999540402205e-05|10000000.0
2001:67c:24cc:31c3::/64|1|#62|61 63 62 51|0.0214999988675117|2.09998379432363e-05|10000000.0
2001:67c:24cc:31c3::/64|1|#41|61 41 51|0.0305000003427267|2.89997115032747e-05|10000000.0
2001:67c:24cc:31c3::/64|1|#41|61 62 51|0.0115000000223517|1.29999416458304e-05|10000000.0
2001:67c:24cc:31c3::/64|1|#41|61 63 62 51|0.0219999980181456|2.19998164538993e-05|10000000.0
2001:67c:24cc:31c3::/64|2|#62|61 41 51|0.00600000005215406|5.99998611505725e-06|16000000.0
2001:67c:24cc:31c3::/64|2|#62|61 62 51|0.00400000018998981|3.99999407818541e-06|20000000.0
2001:67c:24cc:31c3::/64|2|#62|61 63 62 51|0.00700000021606684|6.9999800871301e-06|16000000.0
2001:67c:24cc:31c3::/64|2|#41|61 41 51|0.00499999988824129|4.99999123348971e-06|16000000.0
2001:67c:24cc:31c3::/64|2|#41|61 62 51|0.00500000035390258|4.99998986924766e-06|20000000.0
2001:67c:24cc:31c3::/64|2|#41|61 63 62 51|0.00800000037997961|7.99997269496089e-06|16000000.0
3101::/64|2|#62|41 51|0.00400000018998981|3.99999498768011e-06|16000000.0
3101::/64|2|#62|41 61 62 51|0.105000011622906|6.99997963238275e-06|16000000.0
3101::/64|2|#62|41 61 63 62 51|0.108000010251999|9.99995700112777e-06|16000000.0
3101::/64|2|#41|41 51|0.00300000002607703|2.99999805974949e-06|16000000.0
3101::/64|2|#41|41 61 62 51|0.106000013649464|7.99997269496089e-06|16000000.0
3101::/64|2|#41|41 61 63 62 51|0.109000012278557|1.09999473352218e-05|16000000.0
3101::/64|1|#62|41 51|0.020499998703599|1.89999027497834e-05|10000000.0
3101::/64|1|#62|41 61 62 51|0.12100000679493|2.99996590911178e-05|10000000.0
3101::/64|1|#62|41 61 63 62 51|0.131499990820885|3.89993838325609e-05|10000000.0
3101::/64|1|#41|41 51|0.0199999995529652|1.79999206011416e-05|10000000.0
3101::/64|1|#41|41 61 62 51|0.121500007808208|3.09996285068337e-05|10000000.0
3101::/64|1|#41|41 61 63 62 51|0.131999984383583|3.99993441533297e-05|10000000.0
2001:67c:24cc:31d2::/64|2|#62|62 51|0.0020000000949949|1.99999908545578e-06|20000000.0
2001:67c:24cc:31d2::/64|2|#62|62 61 41 51|0.00800000037997961|7.99997269496089e-06|16000000.0
2001:67c:24cc:31d2::/64|2|#62|62 63 61 41 51|0.0120000010356307|1.19999367598211e-05|16000000.0
2001:67c:24cc:31d2::/64|2|#41|62 51|0.00300000002607703|2.99999715025479e-06|20000000.0
2001:67c:24cc:31d2::/64|2|#41|62 61 41 51|0.00700000021606684|6.9999800871301e-06|16000000.0
2001:67c:24cc:31d2::/64|2|#41|62 63 61 41 51|0.0110000008717179|1.09999473352218e-05|16000000.0
10.203.62.240/30|1|#62|62 51|0.00549999997019768|5.99999475525692e-06|10000000.0
10.203.62.240/30|1|#62|62 61 41 51|0.0364999994635582|3.59994955942966e-05|10000000.0
10.203.62.240/30|1|#62|62 63 61 41 51|0.051999993622303|4.79990449093748e-05|10000000.0
10.203.62.240/30|1|#41|62 51|0.00600000005215406|6.99998872732976e-06|10000000.0
10.203.62.240/30|1|#41|62 61 41 51|0.0359999984502792|3.49995316355489e-05|10000000.0
10.203.62.240/30|1|#41|62 63 61 41 51|0.051499992609024|4.69990918645635e-05|10000000.0
10.203.62.240/30|2|#62|62 51|0.0020000000949949|1.99999908545578e-06|20000000.0
10.203.62.240/30|2|#62|62 61 41 51|0.00800000037997961|7.99997269496089e-06|16000000.0
10.203.62.240/30|2|#62|62 63 61 41 51|0.0120000010356307|1.19999367598211e-05|16000000.0
10.203.62.240/30|2|#41|62 51|0.00300000002607703|2.99999715025479e-06|20000000.0
10.203.62.240/30|2|#41|62 61 41 51|0.00700000021606684|6.9999800871301e-06|16000000.0
10.203.62.240/30|2|#41|62 63 61 41 51|0.0110000008717179|1.09999473352218e-05|16000000.0
2001:67c:24cc:31d3::/64|2|#62|62 51|0.0020000000949949|1.99999908545578e-06|20000000.0
2001:67c:24cc:31d3::/64|2|#62|62 61 41 51|0.00800000037997961|7.99997269496089e-06|16000000.0
2001:67c:24cc:31d3::/64|2|#62|62 63 61 41 51|0.0120000010356307|1.19999367598211e-05|16000000.0
2001:67c:24cc:31d3::/64|2|#41|62 51|0.00300000002607703|2.99999715025479e-06|20000000.0
2001:67c:24cc:31d3::/64|2|#41|62 61 41 51|0.00700000021606684|6.9999800871301e-06|16000000.0
2001:67c:24cc:31d3::/64|2|#41|62 63 61 41 51|0.0110000008717179|1.09999473352218e-05|16000000.0
2001:67c:24cc:31d3::/64|1|#62|62 51|0.00549999997019768|5.99999475525692e-06|10000000.0
2001:67c:24cc:31d3::/64|1|#62|62 61 41 51|0.0364999994635582|3.59994955942966e-05|10000000.0
2001:67c:24cc:31d3::/64|1|#62|62 63 61 41 51|0.051999993622303|4.79990449093748e-05|10000000.0
2001:67c:24cc:31d3::/64|1|#41|62 51|0.00600000005215406|6.99998872732976e-06|10000000.0
2001:67c:24cc:31d3::/64|1|#41|62 61 41 51|0.0359999984502792|3.49995316355489e-05|10000000.0
2001:67c:24cc:31d3::/64|1|#41|62 63 61 41 51|0.051499992609024|4.69990918645635e-05|10000000.0
```

```
10.203.62.0/27|1|#62|62 51|0.00549999997019768|5.99999475525692e-06|10000000.0
10.203.62.0/27|1|#62|62 61 41 51|0.0364999994635582|3.59994955942966e-05|10000000.0
10.203.62.0/27|1|#62|62 63 61 41 51|0.051999993622303|4.79990449093748e-05|10000000.0
10.203.62.0/27|1|#41|62 51|0.00600000005215406|6.99998872732976e-06|10000000.0
10.203.62.0/27|1|#41|62 61 41 51|0.0359999984502792|3.49995316355489e-05|10000000.0
10.203.62.0/27|1|#41|62 63 61 41 51|0.051499992609024|4.69990918645635e-05|10000000.0
2001:67c:24cc:31d2::/64|1|#62|62 51|0.00549999997019768|5.99999475525692e-06|10000000.0
2001:67c:24cc:31d2::/64|1|#62|62 61 41 51|0.0364999994635582|3.59994955942966e-05|10000000.0
2001:67c:24cc:31d2::/64|1|#62|62 63 61 41 51|0.051999993622303|4.79990449093748e-05|10000000.0
2001:67c:24cc:31d2::/64|2|#41|62 61 41 51|0.00700000021606684|6.9999800871301e-06|16000000.0
2001:67c:24cc:31d2::/64|2|#41|62 63 61 41 51|0.0110000008717179|1.09999473352218e-05|16000000.0
10.203.62.240/30|1|#62|62 51|0.00549999997019768|5.99999475525692e-06|10000000.0
10.203.62.240/30|1|#62|62 61 41 51|0.0364999994635582|3.59994955942966e-05|10000000.0
10.203.62.240/30|1|#62|62 63 61 41 51|0.051999993622303|4.79990449093748e-05|10000000.0
10.203.62.240/30|1|#41|62 51|0.00600000005215406|6.99998872732976e-06|10000000.0
10.203.62.240/30|1|#41|62 61 41 51|0.0359999984502792|3.49995316355489e-05|10000000.0
10.203.62.240/30|1|#41|62 63 61 41 51|0.051499992609024|4.69990918645635e-05|10000000.0
10.203.62.240/30|2|#62|62 51|0.0020000000949949|1.99999908545578e-06|20000000.0
10.203.62.240/30|2|#62|62 61 41 51|0.00800000037997961|7.99997269496089e-06|16000000.0
10.203.62.240/30|2|#62|62 63 61 41 51|0.0120000010356307|1.19999367598211e-05|16000000.0
10.203.62.240/30|2|#41|62 51|0.00300000002607703|2.99999715025479e-06|20000000.0
10.203.62.240/30|2|#41|62 61 41 51|0.00700000021606684|6.9999800871301e-06|16000000.0
10.203.62.240/30|2|#41|62 63 61 41 51|0.0110000008717179|1.09999473352218e-05|16000000.0
2001:67c:24cc:31d3::/64|2|#62|62 51|0.0020000000949949|1.99999908545578e-06|20000000.0
2001:67c:24cc:31d3::/64|2|#62|62 61 41 51|0.00800000037997961|7.99997269496089e-06|16000000.0
2001:67c:24cc:31d3::/64|2|#62|62 63 61 41 51|0.0120000010356307|1.19999367598211e-05|16000000.0
2001:67c:24cc:31d3::/64|2|#41|62 51|0.00300000002607703|2.99999715025479e-06|20000000.0
2001:67c:24cc:31d3::/64|2|#41|62 61 41 51|0.00700000021606684|6.9999800871301e-06|16000000.0
2001:67c:24cc:31d3::/64|2|#41|62 63 61 41 51|0.0110000008717179|1.09999473352218e-05|16000000.0
2001:67c:24cc:31d3::/64|1|#62|62 51|0.00549999997019768|5.99999475525692e-06|10000000.0
2001:67c:24cc:31d3::/64|1|#62|62 61 41 51|0.0364999994635582|3.59994955942966e-05|10000000.0
2001:67c:24cc:31d3::/64|1|#62|62 63 61 41 51|0.051999993622303|4.79990449093748e-05|10000000.0
2001:67c:24cc:31d3::/64|1|#41|62 51|0.00600000005215406|6.99998872732976e-06|10000000.0
2001:67c:24cc:31d3::/64|1|#41|62 61 41 51|0.0359999984502792|3.49995316355489e-05|10000000.0
2001:67c:24cc:31d3::/64|1|#41|62 63 61 41 51|0.051499992609024|4.69990918645635e-05|10000000.0
10.203.62.0/27|1|#62|62 51|0.00549999997019768|5.99999475525692e-06|10000000.0
10.203.62.0/27|1|#62|62 61 41 51|0.0364999994635582|3.59994955942966e-05|10000000.0
10.203.62.0/27|1|#62|62 63 61 41 51|0.051999993622303|4.79990449093748e-05|10000000.0
10.203.62.0/27|1|#41|62 51|0.00600000005215406|6.99998872732976e-06|10000000.0
10.203.62.0/27|1|#41|62 61 41 51|0.0359999984502792|3.49995316355489e-05|10000000.0
```

Figure 16: RAE-TID database dump showing the paths to/from ISPs in COMET testbed

As the picture above shows, RAE ends up collecting all the direct and reverse path connecting TID and any other AS.

4.1.1.2 FUN-UC1-002

This test demonstrates how a Content Owner can publish contents in COMET (Content Publication Operation).

Contents in TID ISP belong to the domain tid1.es. To publish a content, the Content Owner has to open the browser and access to the Content Publisher web application (the URL is https//:[2a02:9008:0:1911:0:50:56a3:4e]:8090/Publisher/index.jsf). Once logged on, a Content Name named “tid1.es/test” is created, together with an empty Content Record.

Figure 17: Creation of a Content Name through the Content Publisher web application

In this Content Record, a Content Source is created for a streaming CS located CS-TID3 [2a02:9008:0:1913:0:50:56a3:62] This Content Source is assigned a Pr CoS and the retrieval data can be found in the next capture.

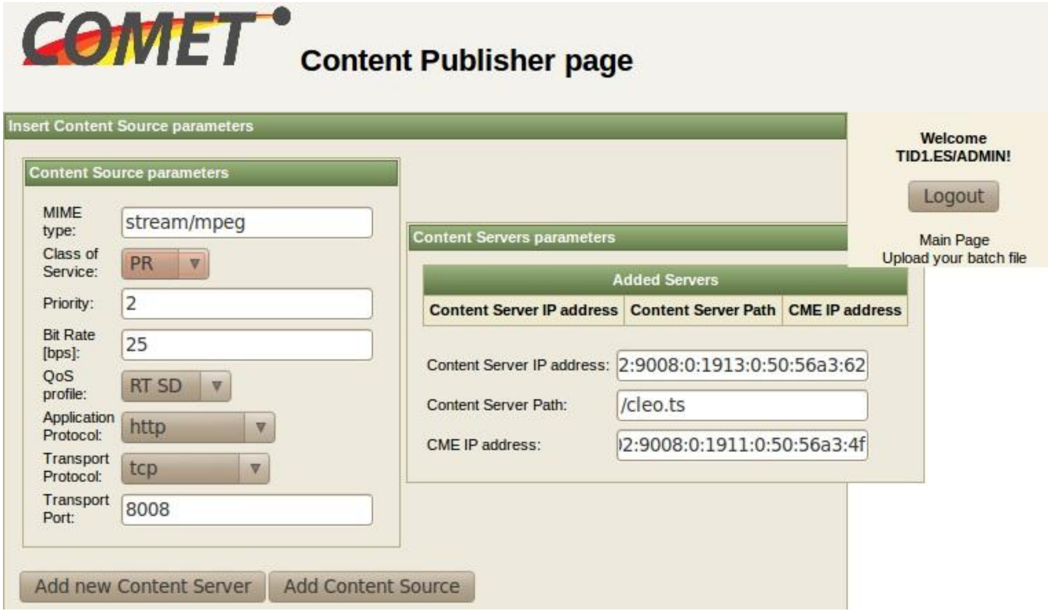


Figure 18: Adding a Content Source in the Content Name

The Content Record is then created and stored in the CRE. By using the same Content Publisher application web, it is easy to check that the date is available and accessible in the CRE..

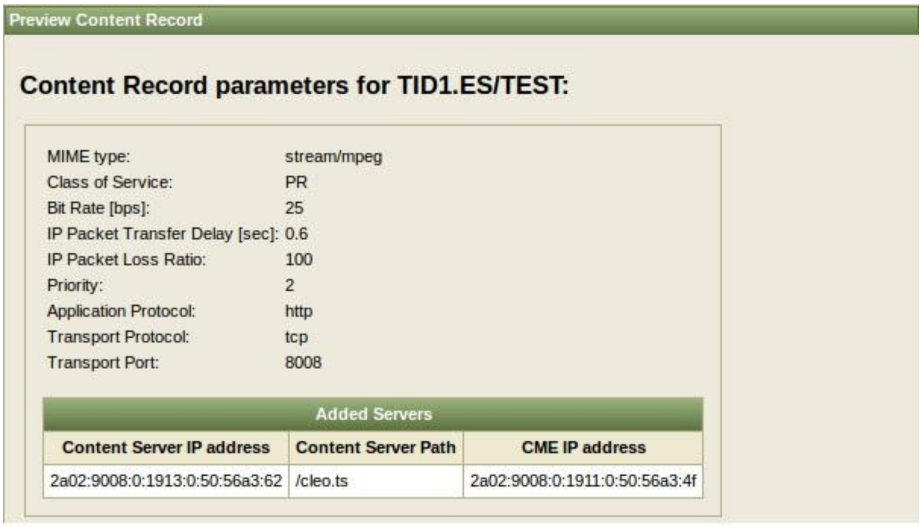


Figure 19: Content Record tid1.es/TEST created

The creation of the Content Record is also reflected in the CRE log as illustrated in the next figure.

```
2012-11-07 14:09:06,877  INFO HandleClient - Content record creation for TID1.ES/TEST.
2012-11-07 14:09:06,878  INFO HandleClient - Reading from file serverdir.txt.
2012-11-07 14:09:06,878  INFO HandleClient - Sending request to server 2A02:9008:0:1911:0:50:56A3:4E:2641
2012-11-07 14:09:21,422  INFO HandleClient - Getting stored record for TID1.ES/TEST.
2012-11-07 14:09:21,422  INFO HandleClient - Reading from file serverdir.txt.
2012-11-07 14:09:21,424  INFO HandleClient - Successful response received by server.
```

Figure 20 CRE log showing the Content Publication Process

4.1.1.3 FUN-UC1-003

This test demonstrates how a CC requests a Content Name from COMET. Specifically, how the CC sends this information to the CME and how this entity decodes and interprets this piece of information.

Following TID testbed description (see section 3.2), CC-TID2 [2a02:9008:0:1912:0:50:56a3:45] is a Pr CoS Content Client and it is used to request tid1.es/test, the content name freshly created in the previous test.

Once the Content Name is written in a web browser in its URL form (comet://tid1.es/test) , the Content Client installed in CC-TID2 sends a message to its local CME requesting for the Content Name. The CME should receive a request and decode it.

As the CME log shows, the CME behaves as expected. The query is received and the Content Name is successfully extracted and decoded.

```
2012-11-21 02:58:53,921 DEBUG DNSDecoder - Received Request = 1353463133921
2012-11-21 02:58:53,921 DEBUG DNSDecoder - Decoding query header...
2012-11-21 02:58:53,921 DEBUG DNSDecoder - 11000001000011
2012-11-21 02:58:53,922 DEBUG DNSDecoder - Read ID= 0C
2012-11-21 02:58:53,922 DEBUG DNSDecoder - 1000000
2012-11-21 02:58:53,922 DEBUG DNSDecoder - byte# = 1 bit# = 7
2012-11-21 02:58:53,922 DEBUG DNSDecoder - byte# = 1 bit# = 6
2012-11-21 02:58:53,922 DEBUG DNSDecoder - Read QR= 0
2012-11-21 02:58:53,922 DEBUG DNSDecoder - Read IT= 1
2012-11-21 02:58:53,922 DEBUG DNSDecoder - 100000000
2012-11-21 02:58:53,922 DEBUG DNSDecoder - Read QDCOUNT= 1
2012-11-21 02:58:53,922 DEBUG DNSDecoder - 0
2012-11-21 02:58:53,922 DEBUG DNSDecoder - Read ANCOUNT= 0
2012-11-21 02:58:53,922 DEBUG DNSDecoder - 0
2012-11-21 02:58:53,922 DEBUG DNSDecoder - 1100
2012-11-21 02:58:53,922 DEBUG DNSDecoder - Read QNAME length= 12
2012-11-21 02:58:53,922 DEBUG DNSDecoder - Decoding query body...
2012-11-21 02:58:53,922 DEBUG DNSDecoder - 1110100011010010110010000110001001011100110010101110011001011110111010001100101011
1001101110100
2012-11-21 02:58:53,922 DEBUG DNSDecoder - Read QNAME= tid1.es/test
2012-11-21 02:58:53,923  INFO DNSHandler - Message received @ Client interface from 2a02:9008:0:1912:cd86:fcca:6016:fe92:6492
7
```

Figure 21: CME log showing the Content Request Process

4.1.1.4 FUN-UC1-004

This test demonstrates how the Name Resolution process works. In short, once the Content Name has been received by the CME, this entity has to carry out two tasks:

- Find out which authoritative CRE stores the associated Content Name by querying the root CRE, in this case, this root CRE is the one deployed at PTL (IPv6 address is [3105::4]) .
- Query the authoritative CRE for the Content Record associated to the Content Name. In this case, the authoritative CRE is the one at TID (CRE-B with IP address 2a02:9008:0:1911:0:50:56a3:4e).

Therefore, CME has to send two queries; one to the CRE root at PLT and another one to the authoritative CRE at TID.

The following CME log dump shows how this element does issue to queries to both CREs.

```
2012-11-21 02:58:53,928 DEBUG HandleClient - Resolution request for naming authority TID1.ES/NA was sent to root CRE with IP
address 3105:0:0:0:0:0:0:4 at port 2641.
2012-11-21 02:58:53,928 DEBUG HandleClient - Before root resolution = 1353463133928
2012-11-21 02:58:53,928 DEBUG HandleClient - 5524979640211030331 T1 7
2012-11-21 02:58:53,940 DEBUG CachedPathsTask - Expired cached paths removed from database.
2012-11-21 02:58:54,260 DEBUG HandleClient - After root resolution = 1353463134260
2012-11-21 02:58:54,260 DEBUG HandleClient - Resolution request for content name tid1.es/test was sent to local CRE with IP a
ddress 2a02:9008:0:1911:0:50:56a3:4e at port 2641.
2012-11-21 02:58:54,260 DEBUG HandleClient - Before authoritative resolution = 1353463134260
2012-11-21 02:58:54,260 DEBUG HandleClient - 5524979640211030331 T2 0
2012-11-21 02:58:54,266 DEBUG HandleClient - After authoritative resolution = 1353463134266
2012-11-21 02:58:54,270 DEBUG Controller - Number of received sources: 1
2012-11-21 02:58:54,270 DEBUG Controller - Number of filtered sources: 1
```

Figure 22: CME log showing the Name Resolution Process

As a final note, CC-TID is a Pr CoS client and the Content Source for tid1.es is also labelled as Pr CoS. Therefore, the CME decides that both CoSs match and that the retrieved Content Source is eligible for further processing.

4.1.1.5 FUN-UC1-005

This test demonstrates how a client CME obtains the status of the CSs in a Content Source being processed following a CC query.

Generally speaking, for a given Content Source a CME has to issue queries to all the remote CMEs managing CSs defined in the Content Server. Those CMEs will in turn query their SNME for the load status and return them to the original CME.

In this case, the Content Source stores a single server located in TID ISP (CS-TID3), so only one SNME hast to be queried, SNME-TID with IP address [2a02:9008:0:1911:0:50:56a3:50]. The SNME has a graphical tool were the load of the Content Servers are depicted in real time.

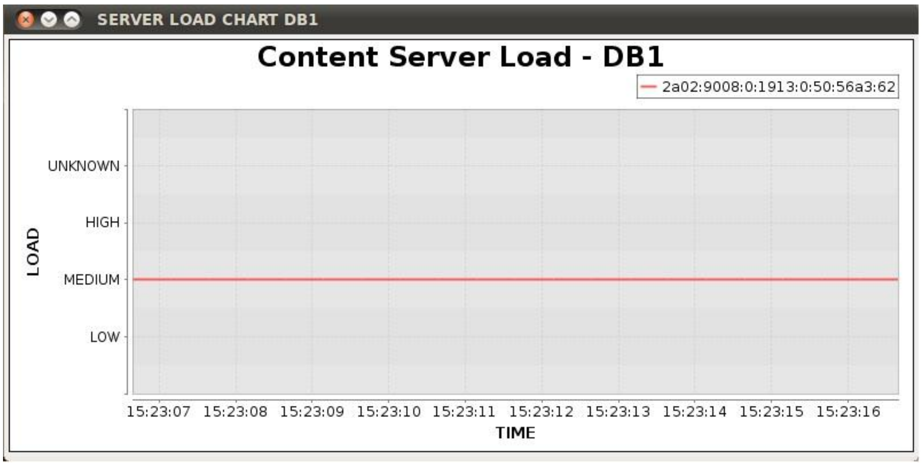


Figure 23: Server loads as monitored by CS-TID3

As reflected in the picture, the load monitored by SNME-TID for CS-TID3 is in on the MEDIUM level. Therefore, The CME must receive this value from its SNME, after querying for CS-TID2.

As the CME log dump shows, this is correctly perform by the CME and the value 2 (internally meaning MEDIUM) is received for the CS-TID3.

```
2012-11-21 02:58:54,294 DEBUG SnmeClient - QueryServerStatus was sent to 2a02:9008:0:1911:0:50:56a3:50 @8888.
2012-11-21 02:58:54,295 DEBUG SnmeHandler - [id: 0x38de5109, /2a02:9008:0:1911:0:50:56a3:4f:60171 => /2a02:9008:0:1911:0:50:5
6a3:50:8888] BOUND: /2a02:9008:0:1911:0:50:56a3:4f:60171
2012-11-21 02:58:54,295 DEBUG SnmeHandler - [id: 0x38de5109, /2a02:9008:0:1911:0:50:56a3:4f:60171 => /2a02:9008:0:1911:0:50:5
6a3:50:8888] CONNECTED: /2a02:9008:0:1911:0:50:56a3:50:8888
2012-11-21 02:58:54,306 DEBUG SnmeHandler - [id: 0x38de5109, /2a02:9008:0:1911:0:50:56a3:4f:60171 :> /2a02:9008:0:1911:0:50:5
6a3:50:8888] DISCONNECTED
2012-11-21 02:58:54,306 DEBUG SnmeHandler - [id: 0x38de5109, /2a02:9008:0:1911:0:50:56a3:4f:60171 :> /2a02:9008:0:1911:0:50:5
6a3:50:8888] UNBOUND
2012-11-21 02:58:54,306 DEBUG SnmeHandler - [id: 0x38de5109, /2a02:9008:0:1911:0:50:56a3:4f:60171 :> /2a02:9008:0:1911:0:50:5
6a3:50:8888] CLOSED
2012-11-21 02:58:54,307 DEBUG ServerAwarenessImpl - After SNME = 1353463134307
2012-11-21 02:58:54,307 DEBUG ServerAwarenessImpl - Returning server load responses to CME handler...
2012-11-21 02:58:54,307 DEBUG ServerAwarenessImpl - Server 2a02:9008:0:1913:0:50:56a3:62 load = 2
```

Figure 24: CME log showing the load of CS-TID3

Since CS load is MEDIUM, this means the server is eligible and processing can go on.

4.1.1.6 FUN-UC1-006

This test demonstrates the Decision Process in the case that CSs and CC are inside the same ISP.

After the operations carried out in the previous test the CME has collected information about paths leading to the involved CSs' ISPs (as informed by the RAE) and the CS statuses. This information is fed into the decision process, which ranks each possible duple CS/Path, according to BW, Packet Loss and Packet delay in the path as well as CS load (a full description of how the decision process works can be found in D3.2 [3]). The duple CS/path with better rates will be the one chosen for content retrieval.

In our case, there is only one possibility the direct internal link between CC-TID2 and CS-TID3 (IP address [2a02:9008:0:1913:0:50:56a3:62]) so this will be duple CS/path identified as having the best rate by the decision process. .

```
2012-11-21 02:58:54,307 DEBUG DecisionMakerImpl - Last server CME response = 1353463134307
2012-11-21 02:58:54,324 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 89.99966, Infinity, 99.83
363, 1.010101, 1.010101
2012-11-21 02:58:54,324 DEBUG DecisionMakerImpl - Server 2a02:9008:0:1913:0:50:56a3:62 and path [51] rate=0.91277903
2012-11-21 02:58:54,324 DEBUG DecisionMakerUtil - Server 2a02:9008:0:1913:0:50:56a3:62 rate=0.91277903
2012-11-21 02:58:54,324 DEBUG Controller - Decision Maker returned the selected record!
2012-11-21 02:58:54,329 DEBUG Controller - Content Client and Server belong to the same domain.
```

Figure 25 CME log showing the Decision Process

Note that if different path were available inside the ISP the would have evaluated and rated.

4.1.1.7 FUN-UC1-007

This test demonstrates the Path Configuration process.

After having found a duple CS/Path as solution, the server CME has to configure the edge CAFE serving the CS with the list of forwarding keys each CAFE in the path has to use to forward the received packets to the next CAFE in the list. Since the paths managed by the CME are in AS format, the path has to be first translated into Key format by the server CME, generally involving the CME of the AS in the path

Since the path found in the previous test is an internal one, all the translation responsibilities fall on CME-TID. According to TID testbed layout (as explained in section 3.2) the CAFEs traversed by a packet transmitted from CS-TID3 to CC-TID2 are CAFE-TID2/CAFE-TIDPT/CAFE-TIDWUT/CAFE-TID3, while the associated path in key format is 0xff, 0xfe, 0xfd. This is the key sequence TID-CME will configure CAFE-TID3, the CAFE managing traffic to/fro CS-TID3

The CME log shows how CME has created the required list of keys (0xff, 0xfe, 0xfd) for the internal path and how this request is sent to CAFE-TID3 [2a02:9008:0:1913:0:50:56a3:55] which is successfully configured as result.

```
2012-11-21 02:58:54,330 DEBUG Controller - CME will create its local KEY_CLIENT and will finish path configuration.
2012-11-21 02:58:54,330 DEBUG PathConfigurationImpl - Configuring path while client and server belong to the same domain.
2012-11-21 02:58:54,340 DEBUG PathConfigurationImpl - Searching for CLIENT KEY in table 3 for 2a02:9008:0:1913:0:50:56a3:55,
2a02:9008:0:1912:0:50:56a3:54, PR
2012-11-21 02:58:54,343  INFO PathConfigurationImpl - FINAL KEY is 0xff, 0xfe, 0xfd
2012-11-21 02:58:54,345 DEBUG CafeConfigurationImpl - Before cafe = 1353463134345
2012-11-21 02:58:54,345 DEBUG CafeConfigurationImpl - 0 T8 1353463134345
2012-11-21 02:58:54,346 DEBUG CafeHandler - [id: 0x7bfea40b] OPEN
2012-11-21 02:58:54,349 DEBUG CafeClient - Generic Request was sent to 2a02:9008:0:1913:0:50:56a3:55 @9999.
2012-11-21 02:58:54,349 DEBUG CafeHandler - [id: 0x7bfea40b, /2a02:9008:0:1911:0:50:56a3:4f:43398 => /2a02:9008:0:1913:0:50:5
6a3:55:9999] BOUND: /2a02:9008:0:1911:0:50:56a3:4f:43398
2012-11-21 02:58:54,349 DEBUG CafeHandler - [id: 0x7bfea40b, /2a02:9008:0:1911:0:50:56a3:4f:43398 => /2a02:9008:0:1913:0:50:5
6a3:55:9999] CONNECTED: /2a02:9008:0:1913:0:50:56a3:55:9999
2012-11-21 02:58:54,358 DEBUG CafeHandler - [id: 0x7bfea40b, /2a02:9008:0:1911:0:50:56a3:4f:43398 :> /2a02:9008:0:1913:0:50:5
6a3:55:9999] DISCONNECTED
2012-11-21 02:58:54,358 DEBUG CafeHandler - [id: 0x7bfea40b, /2a02:9008:0:1911:0:50:56a3:4f:43398 :> /2a02:9008:0:1913:0:50:5
6a3:55:9999] UNBOUND
2012-11-21 02:58:54,358 DEBUG CafeHandler - [id: 0x7bfea40b, /2a02:9008:0:1911:0:50:56a3:4f:43398 :> /2a02:9008:0:1913:0:50:5
6a3:55:9999] CLOSED
2012-11-21 02:58:54,359 DEBUG CafeConfigurationImpl - After cafe = 1353463134359
2012-11-21 02:58:54,359 DEBUG CafeConfigurationImpl - CAFE 2a02:9008:0:1913:0:50:56a3:55 was successfully configured.
2012-11-21 02:58:54,359 DEBUG CafeConfigurationImpl - Stream 302235232:
2012-11-21 02:58:54,362 DEBUG PathConfigurationImpl - CAFE was configured successfully.
```

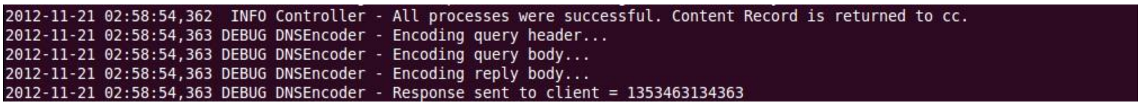
Figure 26: CME log showing the Path Configuration Process

4.1.1.8 FUN-UC1-008

This test demonstrates Content Delivery process

Once the CAFE managing the CS has been successfully configured, the CME will send the connection data to the Content Client, which will launch the default application in the user terminal for that sort of content/protocol. From this moment on, every packet sent by the CS to the CC, will be intercepted by the CAFEs in the path and transmitted to the next one, as defined in the forwarding key list.

In our case, TID CME sends the connection data to CC-TID2 as illustrated in the following CME log dump.



```
2012-11-21 02:58:54,362  INFO Controller - All processes were successful. Content Record is returned to cc.
2012-11-21 02:58:54,363 DEBUG DNSEncoder - Encoding query header...
2012-11-21 02:58:54,363 DEBUG DNSEncoder - Encoding query body...
2012-11-21 02:58:54,363 DEBUG DNSEncoder - Encoding reply body...
2012-11-21 02:58:54,363 DEBUG DNSEncoder - Response sent to client = 1353463134363
```

Figure 27: CME log showing the Content Delivery Process

In CC-TID2 side, VLC player is launched and connects to CS-TID3 by using the following http://[2a02:9008:0:1913:0:50:56a3:62]:8008/cleo.ts, as expected.

Figure 28: CC-TID2 screenshot showing the Content at CS-TID3 being retrieved and the associated URL

4.1.1.9 FUN-UC1-009

This demonstrates how COMET reacts when all CSs in a content source/content record are in HIGH state of occupation.

To perform this test, CS-TID3 [2a02:9008:0:1913:0:50:56a3:62] was shifted to HIGH status. This is reflected on the SNME monitor graphic tool as illustrated in the following picture.

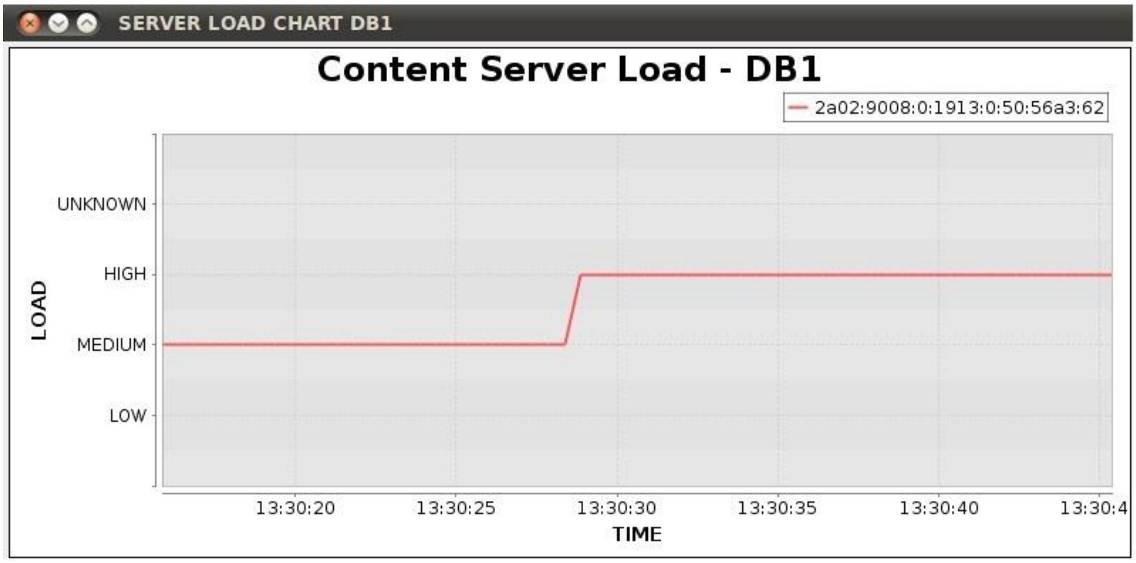


Figure 29: Load in CS-TID3

Owing to CME configuration (see section 3.2.9), the decision algorithm will not be able to find an valid duple CS/Path as solution, so no answer is sent back to CC-TID2 [2a02:9008:0:1912:0:50:56a3:45], as illustrated in the next figure.

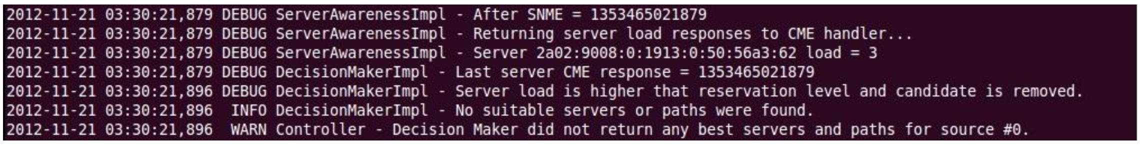


Figure 30: CME log showing how the Decision Process in the CME fails when CSs is overloaded

Since no answer from CME-TID is received, the client asks twice more for the Content Name and eventually desists.

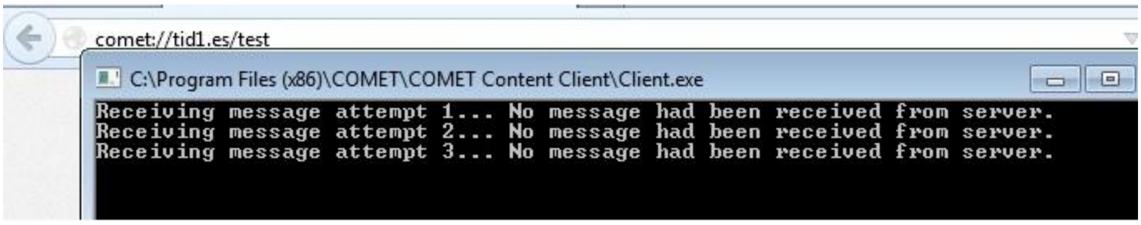


Figure 31: CC Retries after not receiving any answer

4.1.1.10 FUN-UC1-010

The objective of the test is to demonstrate how the system works when a new server is added.

For this case, a new Content Source is going to be added in the Content Record previously created called test. The new server will have the requirement of PR Source for the same content and will be host in cs-pt [3101::3] (set in Primetel ISP). If CC-TID2 would ask for the content, in case no overloaded servers exist, the first candidate in the decision process would be CS-TID3 for being the one with higher priority (as it is set in the Content Record). The second option is going to be the new server added for being the only one left in the Content Record for PR user.

The information we can find in the Content Record updated with the new server is:

Content Record parameters for TID1.ES/TEST:

MIME type:	stream/mpeg
Class of Service:	PR
Bit Rate [bps]:	25
IP Packet Transfer Delay [sec]:	0.6
IP Packet Loss Ratio:	100
Priority:	2
Application Protocol:	http
Transport Protocol:	tcp
Transport Port:	8008

Added Servers

Content Server IP address	Content Server Path	CME IP address
2a02:9008:0:1913:0:50:56a3:62	/cleo.ts	2a02:9008:0:1911:0:50:56a3:4f

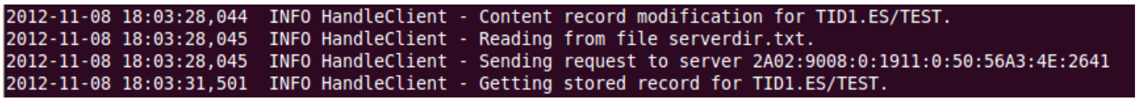
MIME type:	stream/mpeg
Class of Service:	PR
Bit Rate [bps]:	25
IP Packet Transfer Delay [sec]:	0.4
IP Packet Loss Ratio:	0.001
Priority:	1
Application Protocol:	http
Transport Protocol:	tcp
Transport Port:	8008

Added Servers

Content Server IP address	Content Server Path	CME IP address
3101:0:0:0:0:0:0:3	/video_HQ.ts	3105:0:0:0:0:0:0:6

Figure 32: Content Record with a new server added

To check that everything has been done properly, we can confirm it through the CRE log where can be found the process of modifying the Content Record successfully.



```
2012-11-08 18:03:28,044  INFO HandleClient - Content record modification for TID1.ES/TEST.
2012-11-08 18:03:28,045  INFO HandleClient - Reading from file serverdir.txt.
2012-11-08 18:03:28,045  INFO HandleClient - Sending request to server 2A02:9008:0:1911:0:50:56A3:4E:2641
2012-11-08 18:03:31,501  INFO HandleClient - Getting stored record for TID1.ES/TEST.
```

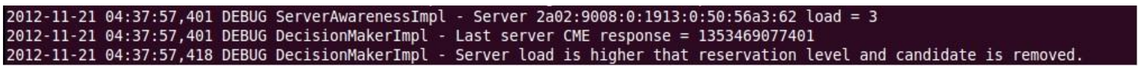
Figure 33: CRE log: Content Record updated

4.1.1.11 FUN-UC1-011

The objective of the test is to demonstrate how the system works when a new user asks for the Content and the new server added is taken into account.

In FUN-UC1-009 CS-TID3 is forced to the HIGH state, a new server is added [FUN-Uc1-010] so if we would ask for the same content with CC-TID2 the result will be different compared to the previous tests [FUN-UC1-003 to FUN-UC1-008]. Once the CME informs that server load in CS-TID3 is high, the content would be retrieved from the new server added in PT (3103::3) for being the only one in the Content Record with the same CoS. The content retrieval will be done between CC-TID2 [2a02:9008:0:1912:0:50:56a3:45] and cs-pt [3101::3]. Accessing to the CME log, the result is what we expected:

- Load in CS-TID3 is HIGH (Server Awareness) so this server is not taken into account.



```
2012-11-21 04:37:57,401 DEBUG ServerAwarenessImpl - Server 2a02:9008:0:1913:0:50:56a3:62 load = 3
2012-11-21 04:37:57,401 DEBUG DecisionMakerImpl - Last server CME response = 1353469077401
2012-11-21 04:37:57,418 DEBUG DecisionMakerImpl - Server load is higher that reservation level and candidate is removed.
```

Figure 34: CME log showing that CS-TID3 is overloaded

- Cs-pt is acceptable in terms of load so the rate of the possible paths to reach the Content is done (Decision Process). After checking that the load of the server is acceptable to retrieve the Content from it, all the possible paths to reach the server are analyzed and rated (this rate is done taking into account different parameters as BW, IP Packet transfer delay…) and every possible path is scored. As can be seen in the CME log above, the paths that connect PT to TID going through WUT are the most appropriate ones (though there are two paths with the same score the one chosen will be the path that crosses the ASs 41, 61, 62 and 51 for being the one with less hops). The rating can be understood better in FUN-UC1-017

```
2012-11-21 04:37:57,699 DEBUG ServerAwarenessImpl - Server 3101:0:0:0:0:0:0:3 load = 2
2012-11-21 04:37:57,731 DEBUG InterCmeHandler - [id: 0x3e97890a, /2a02:9008:0:1911:0:50:56a3:4f:41412 :> /3105:0:0:0:0:0:0:6:
9092] DISCONNECTED
2012-11-21 04:37:57,732 DEBUG InterCmeHandler - [id: 0x3e97890a, /2a02:9008:0:1911:0:50:56a3:4f:41412 :> /3105:0:0:0:0:0:0:6:
9092] UNBOUND
2012-11-21 04:37:57,732 DEBUG InterCmeHandler - [id: 0x3e97890a, /2a02:9008:0:1911:0:50:56a3:4f:41412 :> /3105:0:0:0:0:0:0:6:
9092] CLOSED
2012-11-21 04:37:57,732 DEBUG PathDiscoveryImpl - Received discovered paths for server 3101:0:0:0:0:0:0:3
2012-11-21 04:37:57,733 DEBUG PathDiscoveryImpl - Received proto paths size = 3
2012-11-21 04:37:57,739 DEBUG PathDiscoveryImpl - Returned paths size = 3
2012-11-21 04:37:57,749 DEBUG DecisionMakerImpl - Last server CME response = 1353469077749
2012-11-21 04:37:57,765 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 79.999695, Infinity, 96.9
9981, 0.90707064, 1.010101
2012-11-21 04:37:57,765 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 51] rate=0.90707064
2012-11-21 04:37:57,765 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 59.99977, Infinity, 73.49
985, 1.0020202, 1.010101
2012-11-21 04:37:57,765 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 61, 62, 51] rate=0.91277903
2012-11-21 04:37:57,766 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 49.99981, Infinity, 72.49
9855, 0.9979799, 1.010101
2012-11-21 04:37:57,766 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 61, 63, 62, 51] rate=0.91277903
2012-11-21 04:37:57,766 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.91277903
2012-11-21 04:37:57,766 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.91277903
2012-11-21 04:37:57,766 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.90707064
2012-11-21 04:37:57,766 DEBUG Controller - Decision Maker returned the selected record!
```

Figure 35: CME log showing the Decision Process (rating included)

After the path configuration process the Content is retrieve from the CS expected. As it is shown in the picture below, CC-TID2 retrieves the Content from cs-pt [3101::3]. This retrieval is done through the port 8008 of the server being the Content Source the streaming video: video_HQ.ts (all this is indicated in the Content Record).

Figure 36: Screenshot Content Retrieval from the new server

4.1.1.12 FUN-UC1-012

The objective of the test is to demonstrate how a Content Owner changes the type of CSs distribution of the Content.

For performing this test, the server is modified and for simulating a VoD distribution of the Content an Apache Server has been configured. To change from a streaming distribution to a VoD one, the Content Record is modified because now the port that the server uses is different as well as the path where the Content can be found.

- The port now is the 80/TCP (the one that Apache uses).
- The path is changed too because the video is not played in the server when the CC asks for it as happens in the case of streaming sources; for VoD distribution the video is hosted in a directory of the server.

For doing these changes, the Content Owner access to the Content Publisher and modify all the sources of the Content Record with the new parameters and from this moment, the content can be retrieve using the same Content Name but using a VoD distribution. Following can be found a screenshot of a PR VoD Source of the Content Name that will be access with the same Content Name by the PR Content Client [as shown in FUN-UC1-013].

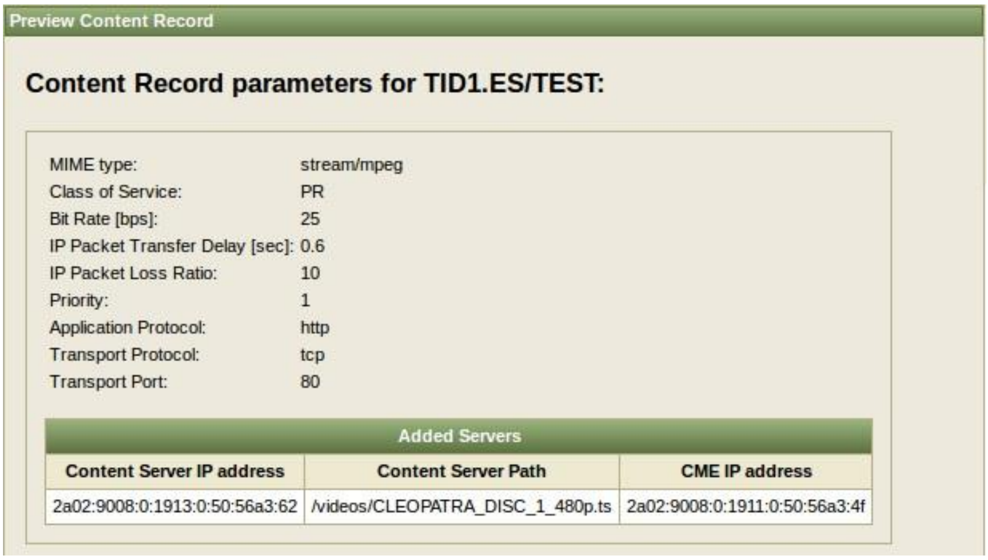


Figure 37: VoD Content Record using the same Content Name

4.1.1.13 FUN-UC1-013

The objective of the test is to demonstrate how VoD Content consumption works using the same Content Record.

After FUN-UC1-012, when a Content Client asks for tid1.es/test Content Name the retrieval has to be performed through a VoD distribution. CC-TID2 has a PR CoS and asks for the Content, all the processes demonstrate from FUN-UC1-003 to FUN-UC1-008 take place and the CS chosen should be a VoD PR one.

As expected, CC-TID3 [2a02:9008:0:1913:0:50:56a3:62] retrieves the Content from a content server with the same CoS and through the port where the Apache is working.

Figure 38: VoD Content Consumption

4.1.1.14 Complex Scenarios

4.1.1.14.1 FUN-UC1-014

The objective of the test is to demonstrate how CCs in different domains can access to the same Content Name.

For performing this test, a PR CC located in WUT asks for the Content hosted in a PR CS in TID domain. The processes followed to retrieve the content from another domain are the same as in FUN-UC1-003 to FUN-UC1-008 but taking into account that the CME located in WUT has to talk to the one in TID with the objective of performing two tasks:

- Knowledge of the load in the server
- Keys configuration for configuring the cafe attached to the server that will offer the Content.

The CC asks for the Content Name and the CRE will inform about the possible Sources that can offer the Content. The CME in WUT is informed about the four possible Sources.

```
2012-11-21 15:26:57,613 DEBUG DNSDecoder - Read QNAME= tid1.es/test
2012-11-21 15:26:57,618  INFO DNSHandler - Message received @ Client interface from 2001:67c:24cc:31c2:0:0:0:26:3929
2012-11-21 15:26:57,626 DEBUG HandleClient - Resolution request for naming authority TID1.ES/NA was sent to root CRE with IP
address 3105:0:0:0:0:0:0:4 at port 2641.
2012-11-21 15:26:57,954 DEBUG HandleClient - Resolution request for content name tid1.es/test was sent to local CRE with IP a
ddress 2a02:9008:0:1911:0:50:56a3:4e at port 2641.
2012-11-21 15:26:58,608 DEBUG Controller - Name resolution time = 990
2012-11-21 15:26:58,623 DEBUG Controller - Number of received sources: 4
2012-11-21 15:26:58,623 DEBUG Controller - Number of filtered sources: 4
2012-11-21 15:26:58,633 DEBUG ServerAwarenessImpl - Sending request for all content servers adjacent to 2a02:9008:0:1911:0:50
:56a3:4f.
```

Figure 39: WUT CME log: Name Resolution Process

The CME in WUT asks for the state of the servers (Server awareness) finding that the server with the same CoS can be chosen because is available for retrieving the Content in terms of load.

```
2012-11-21 15:26:59,131 DEBUG ServerAwarenessImpl - Returning loads for all servers.
2012-11-21 15:26:59,131 DEBUG ServerAwarenessImpl - Server 2a02:9008:0:1913:0:50:56a3:62 load = 2
2012-11-21 15:26:59,131 DEBUG PathDiscoveryImpl - Checking for paths in local cache.
2012-11-21 15:26:59,143 DEBUG PathDiscoveryImpl - Performing longest prefix match for client and server prefix.
2012-11-21 15:26:59,157 DEBUG PathDiscoveryImpl - Found paths in local cache for server 2a02:9008:0:1913:0:50:56a3:62
```

Figure 40: WUT CME log: Decision Process

Once the decision process is done, the CME in WUT detects that Content Client and Content Server belongs to different domains. So the CME creates a local KEY_CLIENT to start the Path Configuration Process to prepare the path for retrieving the Content. From the point of view of the client side the CME in WUT informs about the key of the cafe attached to it and sends a request to the CME of the server side (CME in TID) in order to configure the cafe attached to the server.

```
2012-11-21 15:26:59,166 DEBUG Controller - Decision Maker returned the selected record!
2012-11-21 15:26:59,172 DEBUG Controller - Content Client and Server belong to different domains.
2012-11-21 15:26:59,172 DEBUG Controller - CME will create its local KEY_CLIENT and will start path configuration.
2012-11-21 15:26:59,172 DEBUG PathConfigurationImpl - Processing client side.
2012-11-21 15:26:59,182 DEBUG PathConfigurationImpl - Matching T2 path and cos: 62, 61, PR
2012-11-21 15:26:59,186 DEBUG PathConfigurationImpl - Ingress cafe: 2001:67c:24cc:31c0:0:0:cafe:f
2012-11-21 15:26:59,187 DEBUG PathConfigurationImpl - CLIENT KEY is 0xee
2012-11-21 15:26:59,187 DEBUG Controller - CME will now process server-side CME.
2012-11-21 15:26:59,187 DEBUG PathConfigurationImpl - Preparing request to process server-side CME.
```

Figure 41: WUT CME log: Path Configuration Process client-side

The CME of TID (the one that handles the server) receives the path key that is needed to configure for the retrieval by the CME in WUT, a KEY_SERVER is created and the cafe attached to the server is configured.

```
2012-11-21 17:54:17,252 DEBUG CmeHandler - Request received in server-side CME = 0
2012-11-21 17:54:17,252 DEBUG CmeHandler - Message of type PROCESS_SERVER received @ inter-CME interface from 2001:67c:24cc:3
1c1:0:0:ae:1:41688
2012-11-21 17:54:17,252 DEBUG CmeHandler - path configuration request received = 1353516857252
2012-11-21 17:54:17,255 DEBUG PathConfigurationImpl - PATH KEY for 51, 62, 61, PR was found in local db. SERVER KEY is needed
.
2012-11-21 17:54:17,262 DEBUG PathManagerUtil - Creating SERVER_KEY.
2012-11-21 17:54:17,270 DEBUG PathManagerUtil - SERVER KEY is 0xfd
2012-11-21 17:54:17,270 DEBUG PathConfigurationImpl - KEY_SERVER was successfully created.
2012-11-21 17:54:17,271  INFO PathConfigurationImpl - Final key is [0xee, 0x61, 0xec, 0xfc, 0xfd]
2012-11-21 17:54:17,278 DEBUG CafeConfigurationImpl - Before cafe = 1353516857278
2012-11-21 17:54:17,278 DEBUG CafeConfigurationImpl - 0 T8 26
2012-11-21 17:54:17,280 DEBUG CafeHandler - [id: 0x0596ff94] OPEN
2012-11-21 17:54:17,282 DEBUG CafeClient - Generic Request was sent to 2a02:9008:0:1913:0:50:56a3:55 @9999.
2012-11-21 17:54:17,282 DEBUG CafeHandler - [id: 0x0596ff94, /2a02:9008:0:1911:0:50:56a3:4f:44834 => /2a02:9008:0:1913:0:50:5
6a3:55:9999] BOUND: /2a02:9008:0:1911:0:50:56a3:4f:44834
2012-11-21 17:54:17,282 DEBUG CafeHandler - [id: 0x0596ff94, /2a02:9008:0:1911:0:50:56a3:4f:44834 => /2a02:9008:0:1913:0:50:5
6a3:55:9999] CONNECTED: /2a02:9008:0:1913:0:50:56a3:55:9999
2012-11-21 17:54:17,288 DEBUG CafeHandler - [id: 0x0596ff94, /2a02:9008:0:1911:0:50:56a3:4f:44834 :> /2a02:9008:0:1913:0:50:5
6a3:55:9999] DISCONNECTED
2012-11-21 17:54:17,288 DEBUG CafeHandler - [id: 0x0596ff94, /2a02:9008:0:1911:0:50:56a3:4f:44834 :> /2a02:9008:0:1913:0:50:5
6a3:55:9999] UNBOUND
2012-11-21 17:54:17,288 DEBUG CafeHandler - [id: 0x0596ff94, /2a02:9008:0:1911:0:50:56a3:4f:44834 :> /2a02:9008:0:1913:0:50:5
6a3:55:9999] CLOSED
2012-11-21 17:54:17,289 DEBUG CafeConfigurationImpl - After cafe = 1353516857289
2012-11-21 17:54:17,290 DEBUG CafeConfigurationImpl - CAFE 2a02:9008:0:1913:0:50:56a3:55 was successfully configured.
```

Figure 42: TID CME log: Path Configuration Process server-side

The CME in WUT will configure the rest of the keys from the Content Client until the border of the domain. Now the whole path is configured and once all these Process are done successfully the CC can retrieve the Content.

```
2012-11-21 15:26:59,671 DEBUG PathConfigurationImpl - Server-side CME and path were configured successfully.
2012-11-21 15:26:59,671  INFO Controller - All processes were successful. Content Record is returned to cc.
```

Figure 43: WUT CME log: Content Retrieval

The test is successful so the CC can CC retrieve the content:

Figure 44: Screenshot Content Retrieval with a CC in another domain

4.1.1.14.2 FUN-UC1-015

The objective of the test is to demonstrate how the CME is capable of distinguishing users of different CoS and assigning them to different servers for Content retrieval.

In the Content Record there are three streaming Content Sources one for each CoS and located each one in different domains; PR Source in TID in the host CS-TID3 [2a02:9008:0:1913:0:50:56a3:62], BTBE Source in PT domain in the host cs-pt [3101::3] and BE Source in WUT domain cs_2 [2001:67c:24cc:31d3::242]. We are going to ask for the Content with three different Content Clients in the same domain defined with different CoS; one for each type. CC-TID2 is a PR one, CC-TID4 has a BTBE CoS and CC-TID3 BE respectively. Assuming no Content Server is in high state, each content client should retrieve the content from the content server with the same CoS.

After the test, the result is the one expected, CC-TID2 retrieves the content from CS-TID3:

Figure 45: CC-TID2 Screenshot: Content Retrieval from a PR Content Source

CC-TID4 retrieves the content from cs-pt:

Figure 46: CC-TID4 Screenshot: Content Retrieval from a BTBE Content Source

and CC-TID3 gets the content from cs_2:

Figure 47: CC-TID3 Screenshot: Content Retrieval from a BE Content Source

4.1.1.14.3 FUN-UC1-016

The objective of the test is to demonstrate how the CME manages the CoS rules.

In the test FUN-UC1-015 is demonstrated that using the same Content Name, CCs retrieve the Content from Sources with the same CoS as them. That is:

- PR CC retrieves the content from a PR Content Source
- BTBE CC establishes the communication to retrieve the Content from a BTBE Content Source
- BE gets it from a BE Content Source

But what happens when there is no Source with the same CoS as the CC that asks for the content? To perform this test the same Content Record as in FUN-UC1-015 is used. There are two possible cases:

- The CC has a higher CoS then the available Sources published in the Content Record.

To perform this test CS-TID3 (CoS PR) is forced to be in HIGH state, as well as cs_2 located in WUT. Let us focus in the retrieval of CC-TID2 (PR); when CC-TID2 (PR) asks for the Content no Source PR servers are available for being overloaded so the Content will be retrieved from servers with lower CoS, in our case CC-TID2 will retrieve the content from the BTBE server as CME log shows.

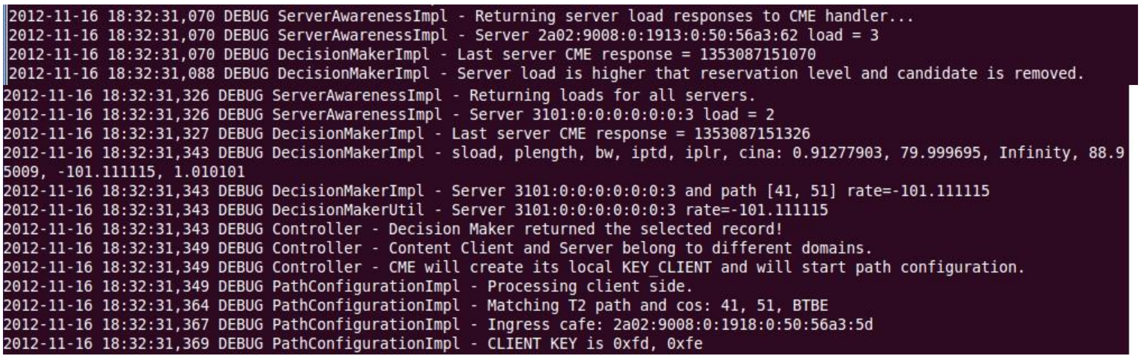

```
2012-11-16 18:32:31,070 DEBUG ServerAwarenessImpl - Returning server load responses to CME handler...
2012-11-16 18:32:31,070 DEBUG ServerAwarenessImpl - Server 2a02:9008:0:1913:0:50:56a3:62 load = 3
2012-11-16 18:32:31,070 DEBUG DecisionMakerImpl - Last server CME response = 1353087151070
2012-11-16 18:32:31,088 DEBUG DecisionMakerImpl - Server load is higher that reservation level and candidate is removed.
2012-11-16 18:32:31,326 DEBUG ServerAwarenessImpl - Returning loads for all servers.
2012-11-16 18:32:31,326 DEBUG ServerAwarenessImpl - Server 3101:0:0:0:0:0:0:3 load = 2
2012-11-16 18:32:31,327 DEBUG DecisionMakerImpl - Last server CME response = 1353087151326
2012-11-16 18:32:31,343 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 79.999695, Infinity, 88.9
5009, -101.111115, 1.010101
2012-11-16 18:32:31,343 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 51] rate=-101.111115
2012-11-16 18:32:31,343 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=-101.111115
2012-11-16 18:32:31,343 DEBUG Controller - Decision Maker returned the selected record!
2012-11-16 18:32:31,349 DEBUG Controller - Content Client and Server belong to different domains.
2012-11-16 18:32:31,349 DEBUG Controller - CME will create its local KEY_CLIENT and will start path configuration.
2012-11-16 18:32:31,349 DEBUG PathConfigurationImpl - Processing client side.
2012-11-16 18:32:31,364 DEBUG PathConfigurationImpl - Matching T2 path and cos: 41, 51, BTBE
2012-11-16 18:32:31,367 DEBUG PathConfigurationImpl - Ingress cafe: 2a02:9008:0:1918:0:50:56a3:5d
2012-11-16 18:32:31,369 DEBUG PathConfigurationImpl - CLIENT KEY is 0xfd, 0xfe
```

Figure 48: CME log showing the Decision and Path Configuration Processes

As CME demonstrate, the first server taken into account is the PR one but is detected that is in HIGH state in terms of load so the server is not an option. After getting again the load of all servers left the decision is made in favor of BTBE server as expected. CC-TID2 retrieves the Content from a BTBE server. So is proved that clients with high CoS can retrieve Contents from servers with lower CoS in case no servers with the same CoS are available.

Figure 49: Screenshot of Content Retrieval

- Sources available have a higher CoS than the CC that asks for the Content.

If CC-TID3 (set as BE CoS) asks for the Content when its corresponding CoS Source is not available, the Content could not be retrieved because clients with low CoS cannot retrieve it from higher CoS servers.

As we expect, the CME indicates that there are no available sources filtered and there is no possibility to award a source where retrieve the Content from:

```
2012-11-16 19:03:34,260 DEBUG Controller - Number of received sources: 2
2012-11-16 19:03:34,260 DEBUG Controller - Number of filtered sources: 0
2012-11-16 19:03:34,260  WARN Controller - There was an error during content resolution process.
```

Figure 50: CME log: Sources have a higher CoS than the CC

Logically, client will not get the Content getting and error for three times:

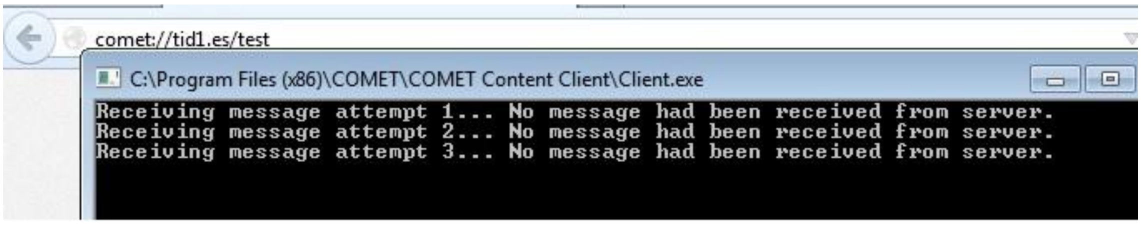


Figure 51: CC Screenshot: no Source can be found

4.1.1.14.4 FUN-UC1-017

The objective of the test is to demonstrate how COMET can assign optimal paths according to the QoS/BW specified in the Content Record.

The only Source published in the Content Record now is located in PT in the server cs-pt [3101::3] and has the following QoS/BW:

MIME type:	stream/mpeg
Class of Service:	PR
Bit Rate [bps]:	25
IP Packet Transfer Delay [sec]:	0.4
IP Packet Loss Ratio:	0.001
Priority:	1
Application Protocol:	http
Transport Protocol:	tcp
Transport Port:	8008

Added Servers

Content Server IP address	Content Server Path	CME IP address
3101:0:0:0:0:0:0:3	/video_HQ.ts	3105:0:0:0:0:0:0:6

Figure 52 Content Record

There are three possible paths to reach the CS:

- AS 41 → AS 51 (Direct Link)
- AS41 → AS 61 → AS 62 → AS 51 (WUT Path)
- AS41 → AS 61 → AS 63 → AS 62 → AS 51 (WUT Path)

For choosing the correct path to configure where the content will be retrieve from the following steps are done:

- Look for the paths with better requirements than the Content Source (candidate paths).
- Do a rating of the candidate paths.
- The one chosen will be the path with higher rate. In case several paths have the same rate, the final path will be the one with fewer hops. (For this reason the path chosen in this test will be the one that crosses WUT going through the Ass 41, 61, 62, and 51.)

For this test, the Decision Process is performed and every possible path that fulfills the requirements of the Content Source is rated. The path with higher score is the one that goes through WUT (rate = 0.91277903). As it is shown in the CME log, keys involved in the content retrieval from TID side are the ones correspondent to CAFE-TIDpt and CAFE-TIDwut (this one is the border cafe that communicates to WUT side).

```
2012-11-21 04:37:57,732 DEBUG PathDiscoveryImpl - Received discovered paths for server 3101:0:0:0:0:0:0:3
2012-11-21 04:37:57,733 DEBUG PathDiscoveryImpl - Received proto paths size = 3
2012-11-21 04:37:57,739 DEBUG PathDiscoveryImpl - Returned paths size = 3
2012-11-21 04:37:57,749 DEBUG DecisionMakerImpl - Last server CME response = 1353469077749
2012-11-21 04:37:57,765 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 79.999695, Infinity, 96.9
9981, 0.90707064, 1.010101
2012-11-21 04:37:57,765 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 51] rate=0.90707064
2012-11-21 04:37:57,765 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 59.99977, Infinity, 73.49
985, 1.0020202, 1.010101
2012-11-21 04:37:57,765 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 61, 62, 51] rate=0.91277903
2012-11-21 04:37:57,766 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 49.99981, Infinity, 72.49
9855, 0.9979799, 1.010101
2012-11-21 04:37:57,766 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 61, 63, 62, 51] rate=0.91277903
2012-11-21 04:37:57,766 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.91277903
2012-11-21 04:37:57,766 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.91277903
2012-11-21 04:37:57,766 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.90707064
2012-11-21 04:37:57,766 DEBUG Controller - Decision Maker returned the selected record!
2012-11-21 04:37:57,771 DEBUG Controller - Content Client and Server belong to different domains.
2012-11-21 04:37:57,772 DEBUG Controller - CME will create its local KEY_CLIENT and will start path configuration.
2012-11-21 04:37:57,772 DEBUG PathConfigurationImpl - Processing client side.
2012-11-21 04:37:57,798 DEBUG PathConfigurationImpl - Matching T2 path and cos: 62, 51, PR
2012-11-21 04:37:57,801 DEBUG PathConfigurationImpl - Ingress cafe: 2a02:9008:0:1919:0:50:56a3:5f
2012-11-21 04:37:57,803 DEBUG PathConfigurationImpl - CLIENT KEY is 0xff, 0xfe
```

Figure 53: CME Log Decision and Path Configuration Processes (rating included)

4.1.1.14.5 FUN-UC1-018

The objective of the test is to demonstrate how COMET will not assign paths whose quality is lower than those defined in the Content Record.

Having the same Content than in the case before, the importance of QoS/BW requirements are demonstrated as follows. The content source has been changed updating a parameter of QoS becoming this more strict. The exact change is higher IP Packet Transfer Delay reqirement, which change from 0.4 to 0.001 seconds. If after the server awareness process, there is no path found that can fulfill this condition, the CME will not be able to perform the rest of the processes to communicate an available source with the client.

When the client asks for tid1.es/test, the resolution is successful and the source is taken into account and filtered as a possible one.

```
2012-11-17 02:12:59,887 DEBUG HandleClient - Resolution request for content name tid1.es/test was sent to local CRE with IP a
ddress 2a02:9008:0:1911:0:50:56a3:4e at port 2641.
2012-11-17 02:12:59,887 DEBUG HandleClient - Before authoritative resolution = 1353114779887
2012-11-17 02:12:59,887 DEBUG HandleClient - -4198126768953706427 T2 1
2012-11-17 02:12:59,896 DEBUG HandleClient - After authoritative resolution = 1353114779896
2012-11-17 02:12:59,900 DEBUG Controller - Number of received sources: 1
2012-11-17 02:12:59,900 DEBUG Controller - Number of filtered sources: 1
2012-11-17 02:12:59,902 DEBUG ServerAwarenessImpl - Sending request for all content servers adjacent to 3105:0:0:0:0:0:0:6.
```

Figure 54: CME log: Content Resolution Process

Its SNME indicates that the load of the server is medium (load=2) so for the moment the server is still available for providing the Content to the client. Now the requirements of the Content Source are compared with the ones of every possible path that can reach the CS. There is no path with better requirements than the Content Source so no path can fulfill the QoS/BW conditions (the ip packet transfer delay of the path does not fulfill what the Content Source needs). There is no path that can satisfy the level of the source so the candidate is removed and no rating can be done.

```
2012-11-17 02:13:00,126 DEBUG ServerAwarenessImpl - Returning loads for all servers.
2012-11-17 02:13:00,126 DEBUG ServerAwarenessImpl - Server 3101:0:0:0:0:0:0:3 load = 2
2012-11-17 02:13:00,132 DEBUG InterCmeHandler - [id: 0x53104f3e, /2a02:9008:0:1911:0:50:56a3:4f:34463 :> /3105:0:0:0:0:0:0:6:
9092] DISCONNECTED
2012-11-17 02:13:00,132 DEBUG InterCmeHandler - [id: 0x53104f3e, /2a02:9008:0:1911:0:50:56a3:4f:34463 :> /3105:0:0:0:0:0:0:6:
9092] UNBOUND
2012-11-17 02:13:00,132 DEBUG InterCmeHandler - [id: 0x53104f3e, /2a02:9008:0:1911:0:50:56a3:4f:34463 :> /3105:0:0:0:0:0:0:6:
9092] CLOSED
2012-11-17 02:13:00,133 DEBUG PathDiscoveryImpl - Received discovered paths for server 3101:0:0:0:0:0:0:3
2012-11-17 02:13:00,133 DEBUG PathDiscoveryImpl - Received proto paths size = 1
2012-11-17 02:13:00,135 DEBUG PathDiscoveryImpl - Returned paths size = 1
2012-11-17 02:13:00,146 DEBUG DecisionMakerImpl - Last server CME response = 1353114780146
2012-11-17 02:13:00,172 DEBUG DecisionMakerImpl - IPTD is higher that reservation level and candidate is removed.
2012-11-17 02:13:00,172  INFO DecisionMakerImpl - No suitable servers or paths were found.
2012-11-17 02:13:00,172  WARN Controller - Decision Maker did not return any best servers and paths for source #0.
2012-11-17 02:13:00,173  WARN Controller - There was an error during content resolution process.
```

Figure 55: CME log Decision Process: paths quality is lower than those defined in the Content Record

Obviously, the client cannot contact to any server to retrieve the Content.

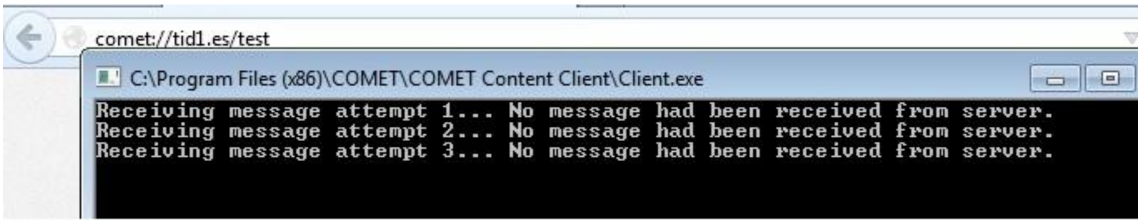


Figure 56: Error in Content Retrieval

4.1.1.14.6 FUN-UC1-019

The objective of the test is to demonstrate how, with every other network parameter the same, the path selection is ruled by servers' load status.

To guarantee the same QoS/BW and confirm that the path follow has just the influence of the load of the servers, the Content Record is going to have two Sources corresponding to two Content Servers of the same LAN with the same CoS as the CC and with the same QoS parameters. That it is:

Content Record parameters for TID1.ES/TEST:

MIME type: stream/mpeg
Class of Service: PR
Bit Rate [bps]: 25
IP Packet Transfer Delay [sec]: 0.6
IP Packet Loss Ratio: 100
Priority: 2
Application Protocol: http
Transport Protocol: tcp
Transport Port: 8008

Added Servers

Content Server IP address	Content Server Path	CME IP address
2a02:9008:0:1913:0:50:56a3:62	/cleo.ts	2a02:9008:0:1911:0:50:56a3:4f
2a02:9008:0:1913:0:50:56a3:63	/cleo.ts	2a02:9008:0:1911:0:50:56a3:4f

Figure 57: Content Record with two Content Sources with the same CoS / QoS / BW

CC-TID2, being a PR Content Client, asks for the Content tid1.es/test, CS-TID4 [2a02:9008:0:1913:0:50:56a3:63] is in high state in terms of load so the Content Source chosen should be the one of the other server. The CME finds out that one candidate to provide the content is overloaded so it is removed as a possible one.

```
2012-11-17 03:05:09,712 DEBUG ServerAwarenessImpl - Returning server load responses to CME handler...
2012-11-17 03:05:09,712 DEBUG ServerAwarenessImpl - Server 2a02:9008:0:1913:0:50:56a3:62 load = 2
2012-11-17 03:05:09,712 DEBUG ServerAwarenessImpl - Server 2a02:9008:0:1913:0:50:56a3:63 load = 3
2012-11-17 03:05:09,713 DEBUG DecisionMakerImpl - Last server CME response = 1353117909713
2012-11-17 03:05:09,729 DEBUG DecisionMakerImpl - Server load is higher that reservation level and candidate is removed.
```

Figure 58: CME log showing the load of the servers (Decision Process)

The CC retrieves the Content from the server expected.

Figure 59: Content retrieval from CS-TID3

If we would force the other server to be overloaded and CS-TID4 goes back to a medium load the server chosen should be the one with IP address 2a02:9008:0:1913:0:50:56a3:63 because of the same reason as explained in the last Content retrieval.

```
2012-11-17 03:16:08,420 DEBUG ServerAwarenessImpl - After SNME = 1353118568420
2012-11-17 03:16:08,420 DEBUG ServerAwarenessImpl - Returning server load responses to CME handler...
2012-11-17 03:16:08,420 DEBUG ServerAwarenessImpl - Server 2a02:9008:0:1913:0:50:56a3:62 load = 3
2012-11-17 03:16:08,420 DEBUG ServerAwarenessImpl - Server 2a02:9008:0:1913:0:50:56a3:63 load = 2
2012-11-17 03:16:08,420 DEBUG DecisionMakerImpl - Last server CME response = 1353118568420
2012-11-17 03:16:08,436 DEBUG DecisionMakerImpl - Server load is higher that reservation level and candidate is removed.
```

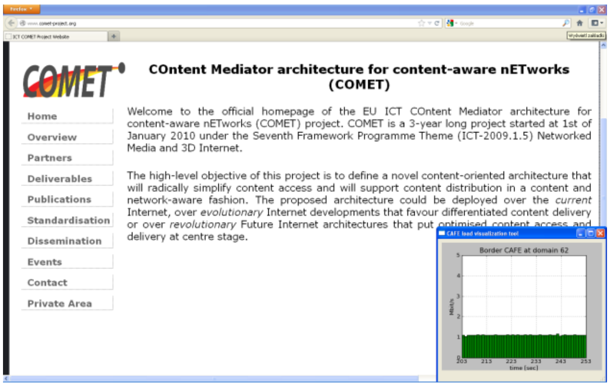
Figure 60: CME log showing the load of the servers (Decision Process)

Figure 61: Content retrieval from CS-TID4

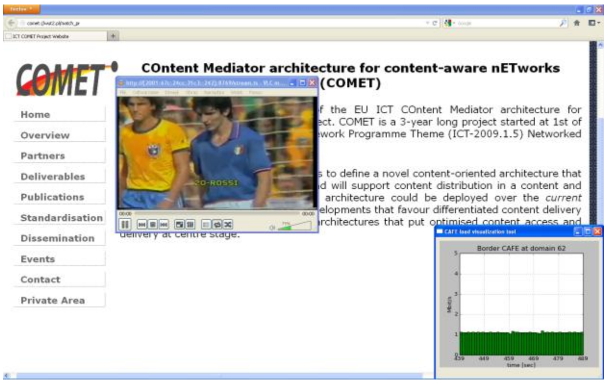
4.1.1.15 Point to multipoint streaming

In section, we present the results of FUN-UC1-020 and FUN-UC1-021 tests that are focused on point to multipoint streaming provided by the Content Streaming Relay (CSR). Moreover, we demonstrate how deployment of the CSR may reduce traffic carried on inter-domain links. We consider scenario where very popular football match is streamed from the content server located in the domain 62, see Figure 1, which shows the federated testbed setup used for this use case. The operator of domain 61 deployed the CSR server to locally serve requests from his customers and offload the inter-domain link between domains 62 and 61. The CSR receives the content stream from the original server and replicate it to local customers.

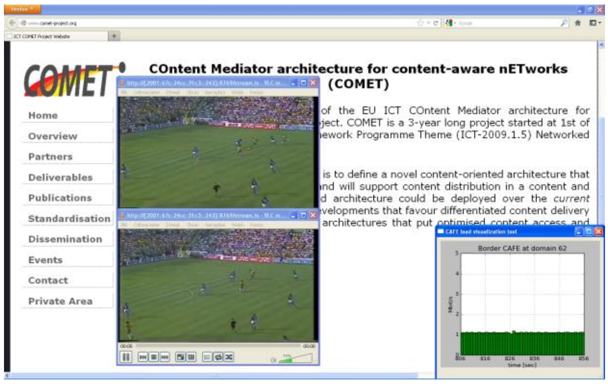
Figure 62 presents the screenshots obtained for consecutive content requests coming from consumers located in the domain 61. Each picture presents view of VLC application and the traffic load measured on the inter-domain link.



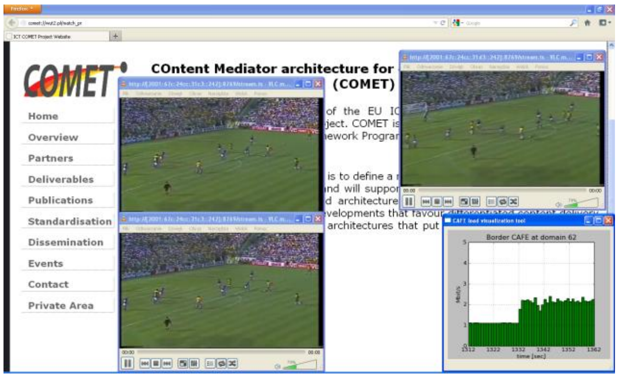
a) before first request



b) after first request



c) after second request



d) after third request

Figure 62: The screenshots obtained for consecutive content requests

1. Situation before first content request

Figure 62a presents the situation observed before arrival of the first content request. Although non VLC application is running yet, we can observe traffic on the inter-domain link. This traffic corresponds to a single stream transferred from the original content server to the CSR server.

2. Situation after first content request

Figure 62b presents situation after the first content request. Once the customer typed the content id wut2.pl/match_pr in a web browser, the browser sends the content request to the CME. Then, the CME selects the CSR server as the best content source and returns CSR URL: http://[2001:67c:24cc:31c3::242]:8769/stream.ts. Finally, the web browser launches the VLC application with the CSR URL and starts playing the content (see figure below).

Figure 63 VLC content after first request

Below, we present details of CME actions performed to serve first content request.

```
2012-11-15 16:56:50,689 DEBUG DNSDecoder - Read QNAME= wut2.pl/match_pr
2012-11-15 16:56:50,693 INFO DNSHandler - Message received @ Client interface from 2001:67c:24cc:31c2:0:0:0:26:4388
2012-11-15 16:56:50,700 DEBUG HandleClient - Resolution request for naming authority WUT2.PL/NA was sent to root CRE with IP address 3105:0:0:0:0:0:0:4 at port 2641.
2012-11-15 16:56:51,016 DEBUG HandleClient - Resolution request for content name wut2.pl/match_pr was sent to local CRE with IP address 2a02:9008:0:1911:0:50:56a3:4e at port 2641.
2012-11-15 16:56:51,607 DEBUG ServerAwarenessImpl - Sending request for all content servers adjacent to 2001:67c:24cc:31c1:0:0:ae:1.
2012-11-15 16:56:51,648 DEBUG ServerAwarenessImpl - Server 2001:67c:24cc:31c3:0:0:0:242 load = 2
2012-11-15 16:56:51,653 DEBUG PathDiscoveryImpl - Found paths in local cache for server 2001:67c:24cc:31c3:0:0:0:242
2012-11-15 16:56:51,654 DEBUG ServerAwarenessImpl - Sending request for all content servers adjacent to 2001:67c:24cc:31d1:0:0:ae:2.
2012-11-15 16:56:51,696 DEBUG ServerAwarenessImpl - Server 2001:67c:24cc:31d3:0:0:0:242 load = 1
2012-11-15 16:56:51,724 DEBUG PathDiscoveryImpl - Found paths in local cache for server 2001:67c:24cc:31d3:0:0:0:242
2012-11-15 16:56:51,737 DEBUG DecisionMakerImpl - Server 2001:67c:24cc:31c3:0:0:0:242 and path [61]rate=0.8888889
2012-11-15 16:56:51,737 DEBUG DecisionMakerImpl - Server 2001:67c:24cc:31d3:0:0:0:242 and path [62, 51, 41, 61] rate=0.22222222
2012-11-15 16:56:51,737 DEBUG DecisionMakerImpl - Server 2001:67c:24cc:31d3:0:0:0:242 and path [62, 63, 61] rate=0.44444445
2012-11-15 16:56:51,737 DEBUG DecisionMakerImpl - Server 2001:67c:24cc:31d3:0:0:0:242 and path [62, 61] rate=0.6666667
2012-11-15 16:56:51,737 DEBUG DecisionMakerUtil - Server 2001:67c:24cc:31c3:0:0:0:242 rate=0.8888889
2012-11-15 16:56:51,737 DEBUG Controller - Decision Maker returned the selected record!
2012-11-15 16:56:51,745 DEBUG Controller - Content Client and Server belong to the same domain.
2012-11-15 16:56:51,755 INFO PathConfigurationImpl - FINAL KEY is 0xee
2012-11-15 16:56:51,818 DEBUG PathConfigurationImpl - CAFE was configured successfully.
2012-11-15 16:56:51,818 INFO Controller - All processes were successful. Content Record is returned to cc.
```

We can distinguish the following processes:

- The name resolution process returns addresses of the CSR server (2001:67c:24cc:31c3:0:0:0:242) and the original server (2001:67c:24cc:31d3:0:0:0:242), located in domain 61 and 62, respectively.
- The server awareness and routing path discovery processes collect information about server load and available routing paths. In the federated testbed, we have four candidates: the CSR server available locally and original server available through three paths: [62, 61], [62, 63, 61] and [62, 51, 41, 61].
- The decision process scores candidates and selects the CSR server as the preferred content source. We can observe that the CSR gets the highest rank among all candidates. Although its load is higher than the original server, it is preferred because it is available locally.
- The CAFE configuration process sets up the edge CAFE to deliver content through selected path.
- Finally, the CME responses to consumer with the CSR URL.

In this case, the traffic on the inter-domain link remains unchanged because the request is served by the CSR server.

3. Situation after second content request

Figure 62c presents situation after the second request. Similar to the previous case, the CME selects the CSR servers as the best content source and returns its URL: http://[2001:67c:24cc:31c3::242]:8769/stream.ts. So, the web browser launches the second instance of VLC application with the CSR URL and starts playing the content (see figure below).

Figure 64 VLC after second request

In this case, the processes performed by CME are similar to the previous case. Again the decision process ranks candidates and selects the CSR as the best content source. The CSR gets the highest rank among four candidates (Server 2001:67c:24cc:31c3:0:0:0:242 rate=0.8888889). The second request is also served locally, so the traffic on the inter-domain link is not increased.

```
012-11-15 16:59:40,132 DEBUG DNSDecoder - Read QNAME= wut2.pl/match_pr
2012-11-15 16:59:40,138 INFO DNSHandler - Message received @ Client interface from
2001:67c:24cc:31c2:0:0:0:26:4394
2012-11-15 16:59:40,141 DEBUG HandleClient - Resolution request for naming authority WUT2.PL/NA
was sent to root CRE with IP address 3105:0:0:0:0:0:0:4 at port 2641.
2012-11-15 16:59:40,459 DEBUG HandleClient - Resolution request for content name wut2.pl/match_pr
was sent to local CRE with IP address 2a02:9008:0:1911:0:50:56a3:4e at port 2641.
2012-11-15 16:59:41,051 DEBUG ServerAwarenessImpl - Sending request for all content servers
adjacent to 2001:67c:24cc:31c1:0:0:ae:1.
2012-11-15 16:59:41,119 DEBUG ServerAwarenessImpl - Server 2001:67c:24cc:31c3:0:0:0:242 load = 2
2012-11-15 16:59:41,125 DEBUG PathDiscoveryImpl - Found paths in local cache for server
2001:67c:24cc:31c3:0:0:0:242
2012-11-15 16:59:41,126 DEBUG ServerAwarenessImpl - Sending request for all content servers
adjacent to 2001:67c:24cc:31d1:0:0:ae:2.
2012-11-15 16:59:41,171 DEBUG ServerAwarenessImpl - Server 2001:67c:24cc:31d3:0:0:0:242 load = 1
2012-11-15 16:59:41,196 DEBUG PathDiscoveryImpl - Found paths in local cache for server
2001:67c:24cc:31d3:0:0:0:242
2012-11-15 16:59:41,210 DEBUG DecisionMakerImpl - Server 2001:67c:24cc:31c3:0:0:0:242 and path
[61] rate=0.8888889
2012-11-15 16:59:41,210 DEBUG DecisionMakerImpl - Server 2001:67c:24cc:31d3:0:0:0:242 and path
[62, 51, 41, 61] rate=0.22222222
2012-11-15 16:59:41,210 DEBUG DecisionMakerImpl - Server 2001:67c:24cc:31d3:0:0:0:242 and path
[62, 63, 61] rate=0.44444445
2012-11-15 16:59:41,210 DEBUG DecisionMakerImpl - Server 2001:67c:24cc:31d3:0:0:0:242 and path
[62, 61] rate=0.6666667
2012-11-15 16:59:41,210 DEBUG DecisionMakerUtil - Server 2001:67c:24cc:31c3:0:0:0:242
rate=0.8888889
2012-11-15 16:59:41,211 DEBUG Controller - Decision Maker returned the selected record!
2012-11-15 16:59:41,221 DEBUG Controller - Content Client and Server belong to the same domain.
2012-11-15 16:59:41,233 INFO PathConfigurationImpl - FINAL KEY is 0xee
2012-11-15 16:59:41,286 DEBUG PathConfigurationImpl - CAFE was configured successfully.
2012-11-15 16:59:41,287 INFO Controller - All processes were successful. Content Record is
returned to cc.
```

4. Situation after third request

Figure 62d presents situation after third request. Now, the CME selects the original server because CSR server went into high load conditions. The CME returns URL of the origin content server http://[2001:67c:24cc:31d3::242]:8769/stream.ts and web browser launches third instance of VLC application and starts playing the content (see figure below).

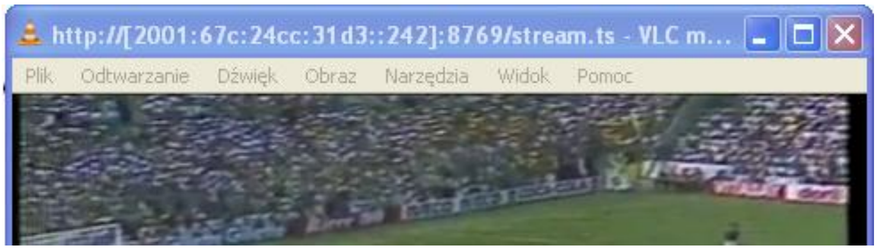


Figure 65 VLC content after third request

Below we present details of CME actions performed to serve third customer request.

```
2012-11-15 17:06:02,779 DEBUG DNSDecoder - Read QNAME= wut2.pl/match_pr
2012-11-15 17:06:02,785 INFO DNSHandler - Message received @ Client interface from 2001:67c:24cc:31c2:0:0:0:26:4396
2012-11-15 17:06:02,795 DEBUG HandleClient - Resolution request for naming authority WUT2.PL/NA was sent to root CRE with IP address 3105:0:0:0:0:0:0:4 at port 2641.
2012-11-15 17:06:03,116 DEBUG HandleClient - Resolution request for content name wut2.pl/match_pr was sent to local CRE with IP address 2a02:9008:0:1911:0:50:56a3:4e at port 2641.
2012-11-15 17:06:03,706 DEBUG ServerAwarenessImpl - Sending request for all content servers adjacent to 2001:67c:24cc:31c1:0:0:ae:1.
2012-11-15 17:06:03,757 DEBUG ServerAwarenessImpl - Server 2001:67c:24cc:31c3:0:0:0:242 load = 3
2012-11-15 17:06:03,781 DEBUG PathDiscoveryImpl - Found paths in local cache for server 2001:67c:24cc:31c3:0:0:0:242
2012-11-15 17:06:03,782 DEBUG ServerAwarenessImpl - Sending request for all content servers adjacent to 2001:67c:24cc:31d1:0:0:ae:2.
2012-11-15 17:06:03,853 DEBUG ServerAwarenessImpl - Server 2001:67c:24cc:31d3:0:0:0:242 load = 1
2012-11-15 17:06:03,885 DEBUG PathDiscoveryImpl - Found paths in local cache for server 2001:67c:24cc:31d3:0:0:0:242
2012-11-15 17:06:03,896 DEBUG DecisionMakerImpl - Server load is higher that reservation level and candidate is removed.
2012-11-15 17:06:03,896 DEBUG DecisionMakerImpl - Server 2001:67c:24cc:31d3:0:0:0:242 and path [62, 51, 41, 61] rate=0.22222222
2012-11-15 17:06:03,896 DEBUG DecisionMakerImpl - Server 2001:67c:24cc:31d3:0:0:0:242 and path [62, 63, 61] rate=0.44444445
2012-11-15 17:06:03,897 DEBUG DecisionMakerImpl - Server 2001:67c:24cc:31d3:0:0:0:242 and path [62, 61] rate=0.6666667
2012-11-15 17:06:03,897 DEBUG DecisionMakerUtil - Server 2001:67c:24cc:31d3:0:0:0:242 rate=0.6666667
2012-11-15 17:06:03,897 DEBUG Controller - Decision Maker returned the selected record!
2012-11-15 17:06:03,899 DEBUG Controller - Content Client and Server belong to different domains.
2012-11-15 17:06:03,900 DEBUG PathConfigurationImpl - Processing client side.
2012-11-15 17:06:03,923 DEBUG PathConfigurationImpl - CLIENT KEY is 0xee
2012-11-15 17:06:03,923 DEBUG Controller - CME will now process server-side CME.
2012-11-15 17:06:03,934 DEBUG InterCmeClient - Generic Request was sent to 2001:67c:24cc:31d1:0:0:ae:2 @9092.
2012-11-15 17:06:04,075 DEBUG PathConfigurationImpl - Server-side CME and path were configured successfully.
2012-11-15 17:06:04,075 INFO Controller - All processes were successful. Content Record is returned to cc.
```

In this case, the processes performed by CME are slightly different. Again, the name resolution process responses with two servers: the CSR and the original server. However, the server status process reports high load on the CSR server, so the decision process selects the original server with the shortest path [62,61] as the preferred solution (Server 2001:67c:24cc:31d3:0:0:0:242 rate=0.6666667). In this case, the path configuration process goes to the CME located in domain 62 to configure the edge cafe. Finally, the CME sends answer to consumer with URL of the origin content server http://[2001:67c:24cc:31d3::242]:8769/stream.ts .

In Figure 62d, we observe increased load on the inter-domain link. This happens because the third content request is served by the original content server located in domain 62 and traffic goes through routing path [62, 61].

The performed tests confirmed that the CSR server correctly supports point to multipoint streaming. Moreover, it allows reducing traffic on inter-domain links proportionally to the number of requests served locally by the CSR.

4.1.2 P2P Offloading (Use Case 4)

4.1.2.1 FUN-UC4-001

The objective of the test is to demonstrate how RAEs gather information about the paths linking the different ISPs (Routing Awareness).

As in FUN-UC1-001 the RAE gather info about the existing paths exchanging information with RAEs of neighbor ISPs knowing how to reach the other domains is checked. In this test, the CME stored in its database this information is verified.

4.1.2.2 FUN-UC4-002

The objective of the test is to demonstrate how a Content Owner distributes a Content publishing a Content Record with the number of Content Sources describing the characteristics and requirement of the streaming Servers.

Content Record tid1.es/TEST_P2P is created with two sources, a Premium streaming one (high priority) and another Best Effort P2P (low priority).

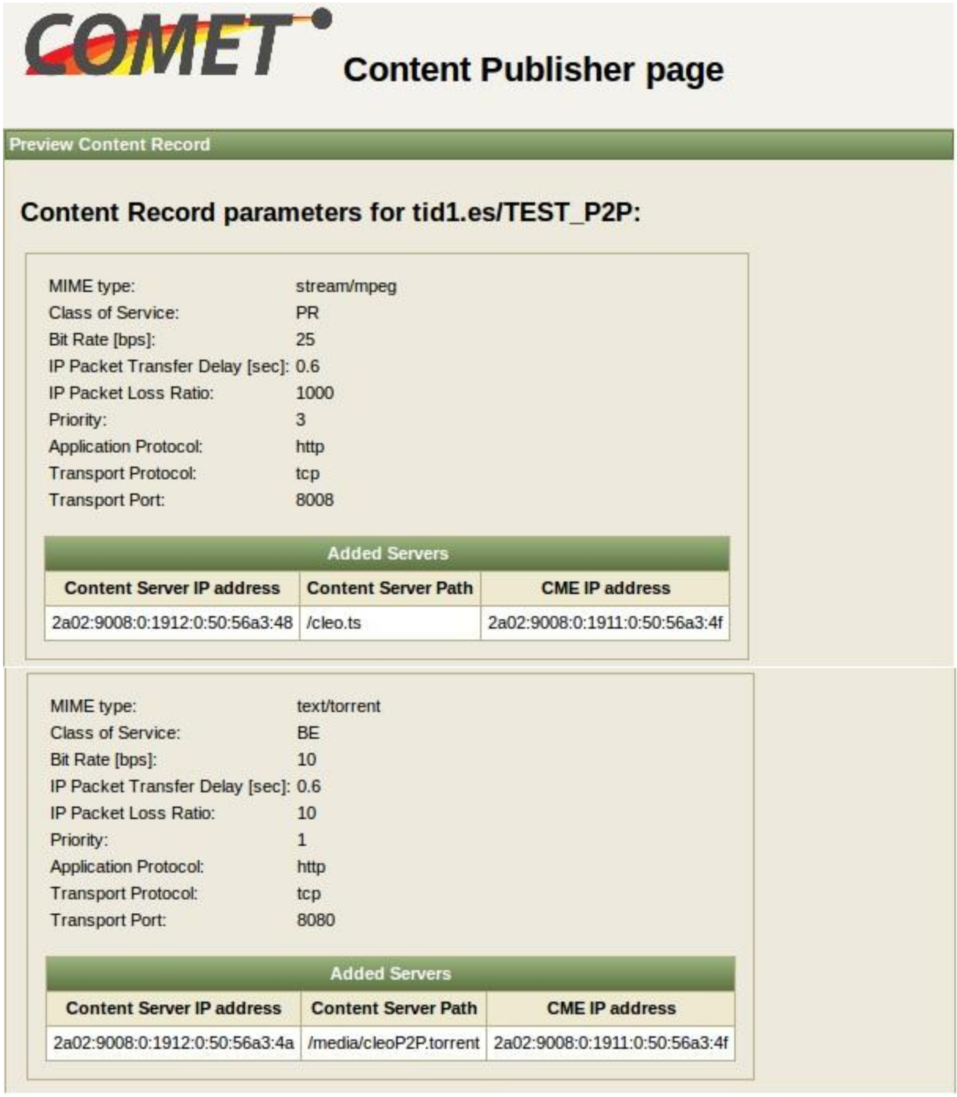


Figure 66: Content Record tid1.es/TEST_P2P

The new Content Record created is stored in the CRE as it was expected.

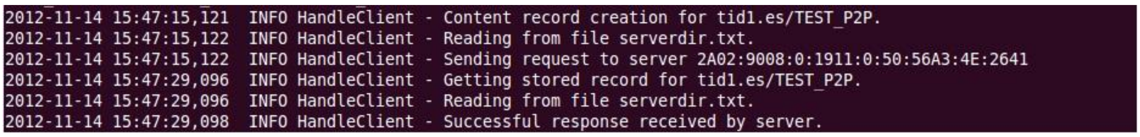


Figure 67: CRE log showing the Content Record TEST_P2P stored

4.1.2.3 FUN-UC4-003

The objective of the test is to demonstrate how an end user requests the Content and retrieves the Content from the most appropriate CS (Streaming Distribution).

Once path are stored at CME [FUN-UC4-001] and the Content is published [FUN-UC4-002]; CC-TID3 [2a02:9008:0:1913:0:50:56a3:61] having a Premium CoS requests the Content through the new Content Name tid1.es/TEST_P2P. The Content should be retrieved from the PR Streaming Source for being the one with the same CoS (high priority); for this reason, the content should be retrieved from CS-TID2 [2a02:9008:0:1912:0:50:56a3:48] and as described in FUN-UC1-004 to FUN-UC1-008 processes Name Resolution, Decision Process, Path Configuration and Content Delivery are done and CC-TID3 can receive the content.

As expected, CME goes through all the processes: Name Resolution (tid1.es/test_p2P), decision process (the server chosen is CS-TID2), path configuration (edge cafe attached to the server is configured), content delivery (Content retrieval from CS-TID2).

```
2012-11-15 12:53:40,897 DEBUG HandleClient - After root resolution = 1352980420897
2012-11-15 12:53:40,897 DEBUG HandleClient - Resolution request for content name tid1.es/test_p2p was sent to loca
l CRE with IP address 2a02:9008:0:1911:0:50:56a3:4e at port 2641.
2012-11-15 12:53:40,897 DEBUG HandleClient - Before authoritative resolution = 1352980420897
2012-11-15 12:53:40,897 DEBUG HandleClient - -3177853719008362378 T2 0
2012-11-15 12:53:40,906 DEBUG HandleClient - After authoritative resolution = 1352980420906
2012-11-15 12:53:40,911 DEBUG Controller - Number of received sources: 2
2012-11-15 12:53:40,911 DEBUG Controller - Number of filtered sources: 2
2012-11-15 12:53:40,914 DEBUG ServerAwarenessImpl - Sending request for all content servers adjacent to 2a02:9008:
0:1911:0:50:56a3:4f.
2012-11-15 12:53:40,958 DEBUG ServerAwarenessImpl - After SNME = 1352980420958
2012-11-15 12:53:40,959 DEBUG ServerAwarenessImpl - Returning server load responses to CME handler...
2012-11-15 12:53:40,959 DEBUG ServerAwarenessImpl - Server 2a02:9008:0:1912:0:50:56a3:48 load = 2
2012-11-15 12:53:40,959 DEBUG DecisionMakerImpl - Last server CME response = 1352980420959
2012-11-15 12:53:40,993 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 89.99966, Infi
nity, 99.7498, 1.0090909, 1.010101
2012-11-15 12:53:40,994 DEBUG DecisionMakerImpl - Server 2a02:9008:0:1912:0:50:56a3:48 and path [51] rate=0.912779
03
2012-11-15 12:53:40,994 DEBUG DecisionMakerUtil - Server 2a02:9008:0:1912:0:50:56a3:48 rate=0.91277903
2012-11-15 12:53:40,994 DEBUG Controller - Decision Maker returned the selected record!
2012-11-15 12:53:40,999 DEBUG Controller - Content Client and Server belong to the same domain.
2012-11-15 12:53:40,999 DEBUG Controller - CME will create its local KEY_CLIENT and will finish path configuration
.
2012-11-15 12:53:40,999 DEBUG PathConfigurationImpl - Configuring path while client and server belong to the same
domain.
2012-11-15 12:53:41,015 DEBUG PathConfigurationImpl - Searching for CLIENT KEY in table 3 for 2a02:9008:0:1912:0:5
0:56a3:54, 2a02:9008:0:1913:0:50:56a3:55, PR
2012-11-15 12:53:41,018  INFO PathConfigurationImpl - FINAL KEY is 0xfd, 0xfe, 0xff
2012-11-15 12:53:41,055 DEBUG PathConfigurationImpl - CAFE was configured successfully.
2012-11-15 12:53:41,055  INFO Controller - All processes were successful. Content Record is returned to cc.
2012-11-15 12:53:41,056 DEBUG DNSEncoder - Encoding query header...
2012-11-15 12:53:41,056 DEBUG DNSEncoder - Encoding query body...
2012-11-15 12:53:41,056 DEBUG DNSEncoder - Encoding reply body...
2012-11-15 12:53:41,056 DEBUG DNSEncoder - Response sent to client = 1352980421056
```

Figure 68: CME log showing Name Resolution, Decision, Path Configuration and Content Delivery processes

CC-TID3 receives the Content from the chosen content server as expected.

Figure 69: Content Retrieval from a streaming distribution

4.1.2.4 FUN-UC4-004

The objective of the test is to demonstrate what happens when load is high in streaming servers.

To test it, Content Server PR is overloaded so the SNME detects that the load of this server is HIGH. If the same CC as in FUN-UC4-003 asks for the same Content with the same Content Name, the content should be retrieved from a lower priority server, in this case, the source chosen would be the P2P distribution one because the server with the same CoS as the Content Client that request the content is overloaded and cannot offer the content.

As it is shown, CS-TID2 is overloaded:

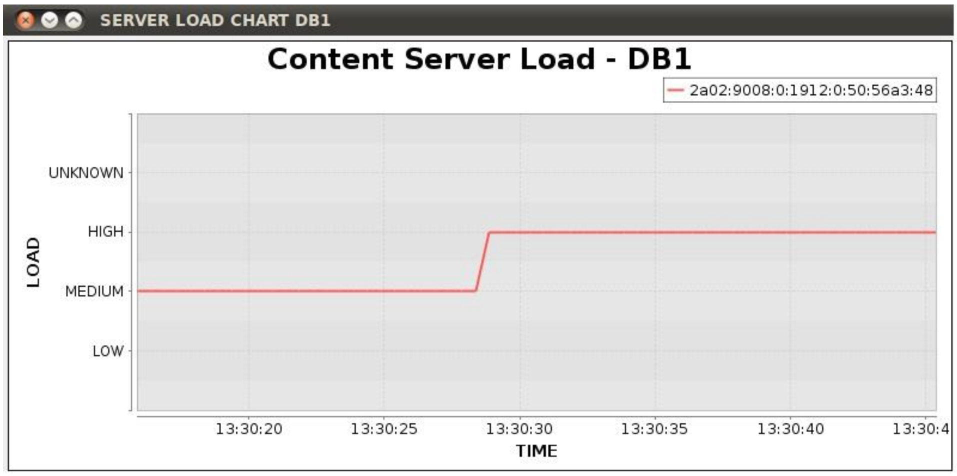


Figure 70: CS-TID2 load

When CC-TID3 requests the Content the server selected is cs-p2p [2a02:9008:0:1912:0:50:56a3:4a] and a P2P application is launched and the content is downloaded in the client.

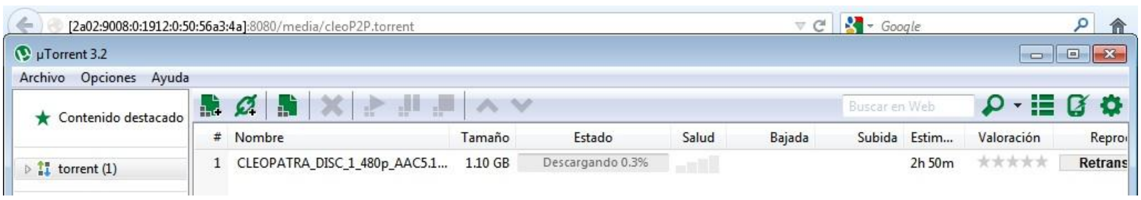


Figure 71: Content Retrieval screenshot from a P2P distribution

4.2 Performance Tests

4.2.1 CRE Tests

4.2.1.1 PER-CRE-001

A large number of records were created by applying COMET means for content publication, i.e. the CP web interface. This test was performed to confirm that there are no COMET imposed limitations to the maximum number of records creatable. Records were created logarithmically i.e. 1000, 10000, 100000 and 1000000 records, using the CP stress tester by running multiple stress test processes and multiple Content publishers (Figure 72). E.g. for establishing 100000 records, 4 CPs were deployed with the below stress test configuration in each CP virtual machine:

 # base part of CP uri
target_base: 'http://localhost:8090/Publisher/'
 # CP login
target_login: 'COMET1'
 # CP password
target_password: 'COMET1'
 # Log directory (must exist)
log_dir: '/root/test_log'
 # Report directory (must exist)
report_dir: '/root/test_report'
 # Number of tests per process
reg_num: 2500
 # Number of processes
proc_num: 10

The results proved that the CRE could store more than 1,000,000 records and effectively the limitations can only be imposed by the system's memory capacity most like caused by the SQL database system. The results were approximately linear with an average of 1200Kbytes per 1000 records as shown below:

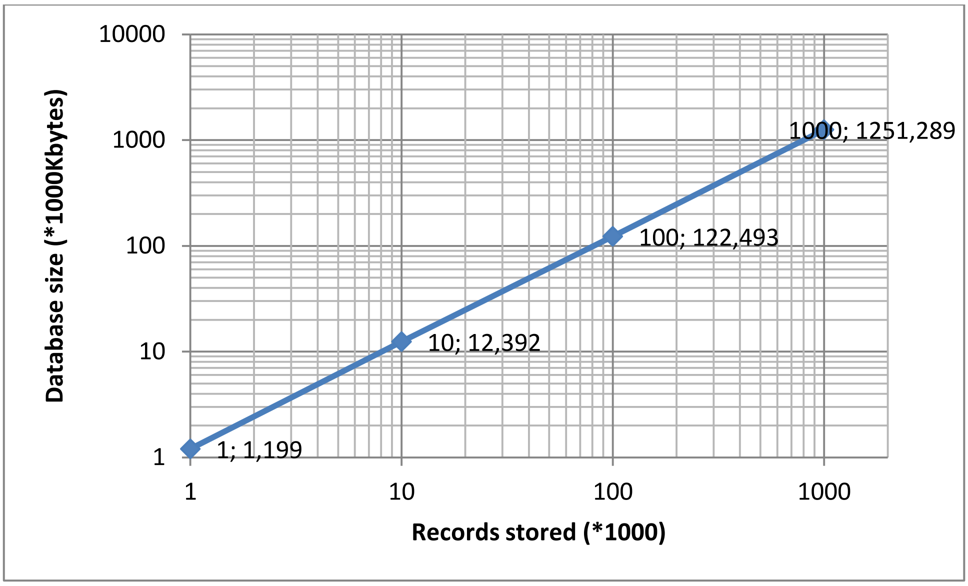


Figure 72: Recorded memory allocation vs no. of CRE records

4.2.1.2 PER-CRE-002-005 - Tests on CRE-CME interface

Figure 73 shows the CPU and Memory consumption of the authoritative CRE when contacted by the CME stress tester where the request rate is programmable. While increasing the request rate from 1 upto 20 requests per second the Memory Consumption remains more or less constant at 20% whule the CPU is occupied between 11-22%.

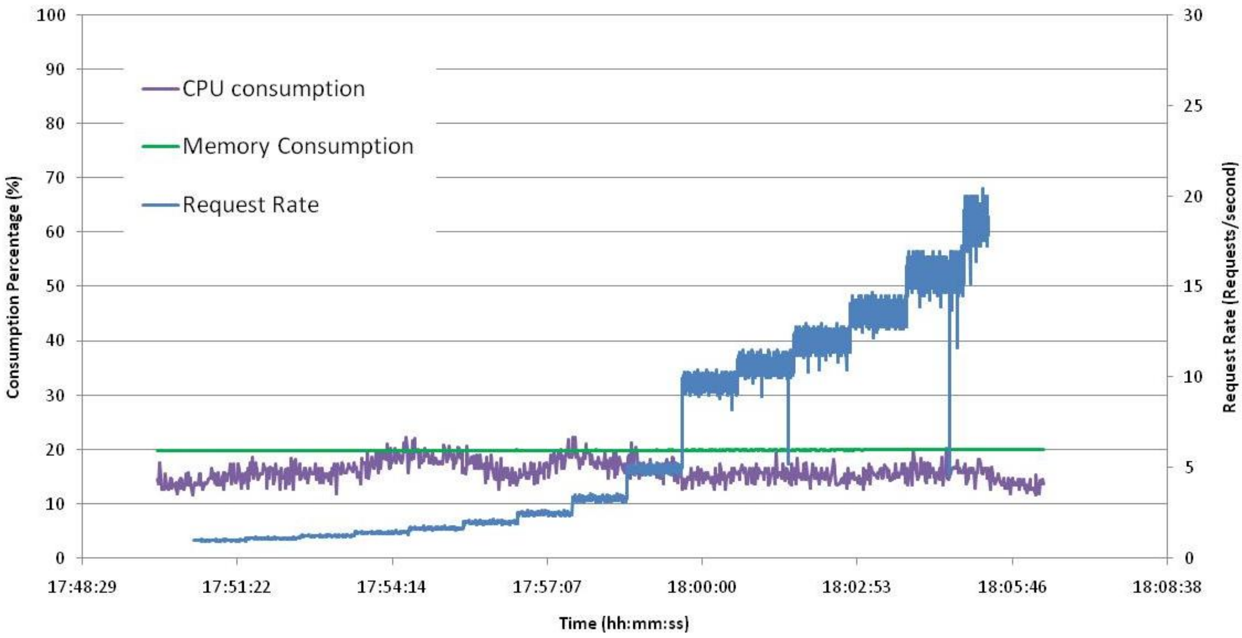


Figure 73 Authoritative CRE Memory and CPU consumption VS Content Request Rate

The corresponding response time is shown in Figure 74 more or less consistent between 2-8ms hence satisfying the expected response time.

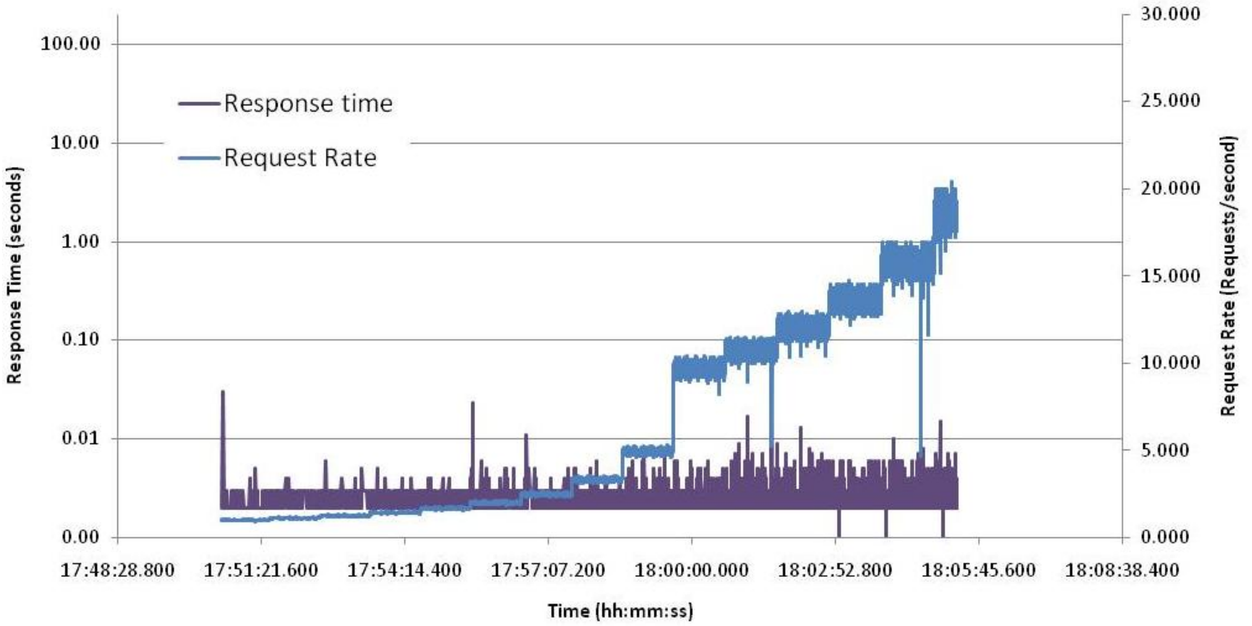


Figure 74 CRE Response Time VS CME Request Rate

In Figure 75 we increased the CME Request Rate to 1000 requests per second. It was noticed that the actual request rate at the CRE was at a value between 100-500 requests per second influenced by TCP to avoid congestion and ensure 100% delivery.

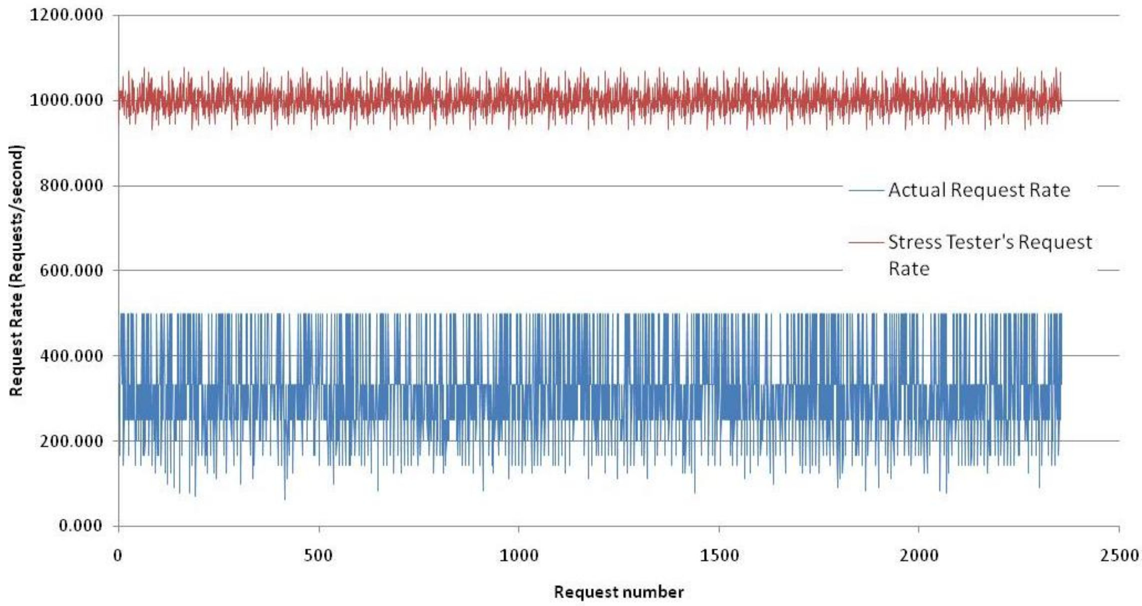


Figure 75 CME Maximum Request Rate to Authoritative CRE

Figure 76 shows the authoritative CRE response time against the CME request rate. It shows how although the request rate was increased from 1 up to close 1000 requests per second the response more or less remains constant at under 10ms, a satisfactory value.

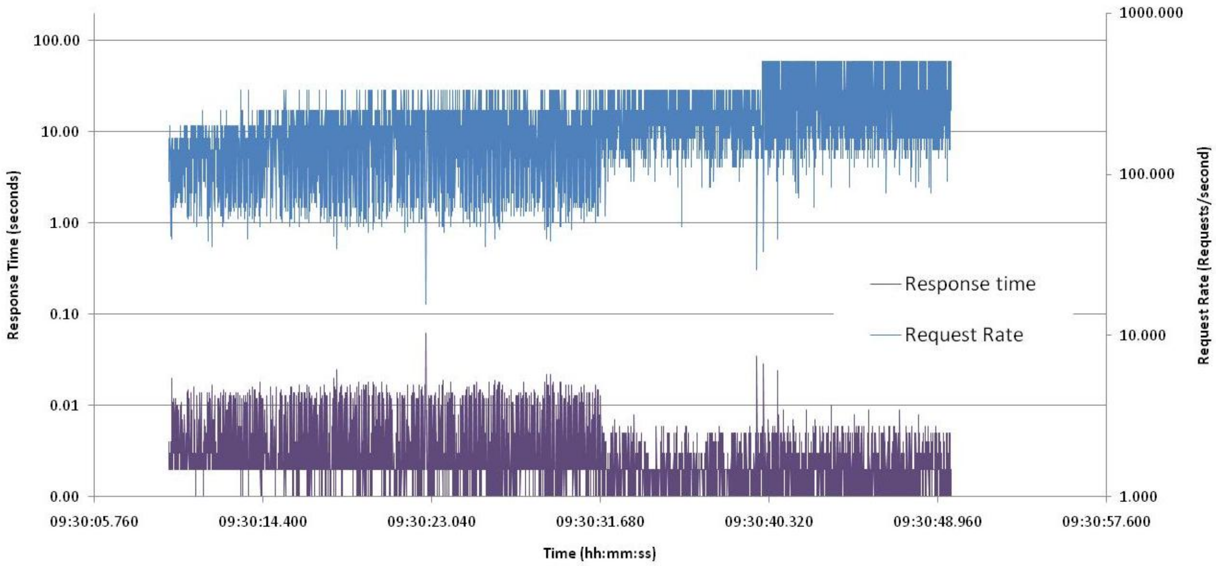


Figure 76 Authoritative CRE Response Time VS CME Request Rate

The tests are repeated but this time testing the Root CRE shown in Figure 77. The test is executed at 500 and 1000 requests per second from the CME to the Root CRE using the developed stress tester with the actual request rate being controlled by the TCP protocol at a value close but below this. The CPU and Memory remain more or less the same for this CRE as in Figure 73.

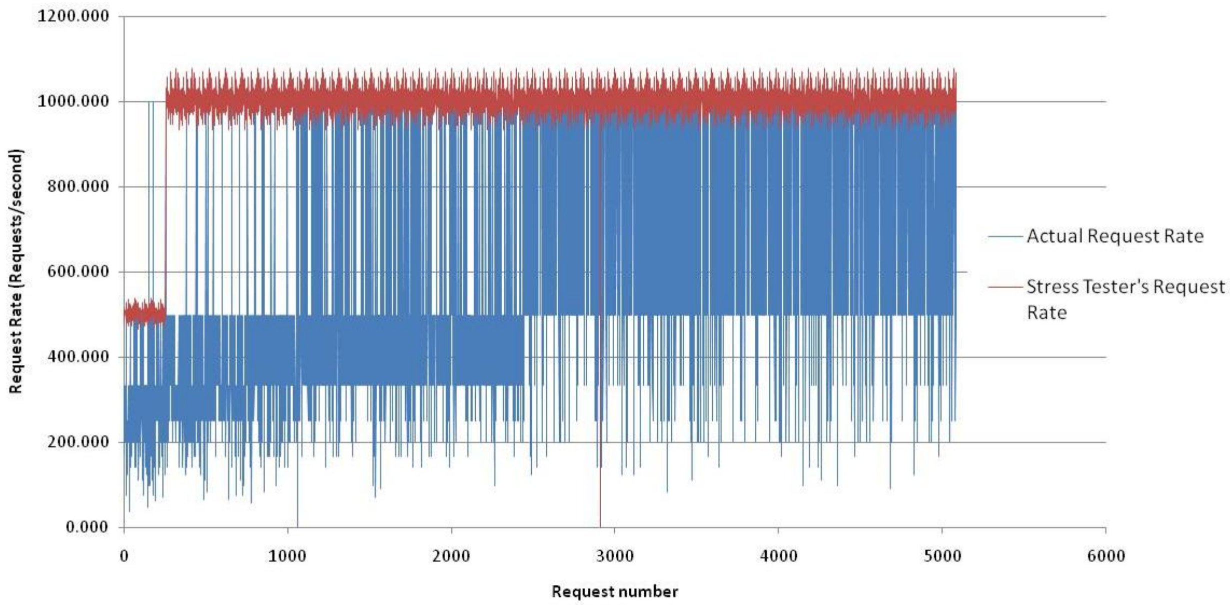


Figure 77 Root CRE Maximum Request Rate

Figure 78 shows the response time of the Root CRE which is mainly below 10ms with the majority of response times ranging between 1-4ms which is close to the expected value even for rates close to 1000 requests per second.

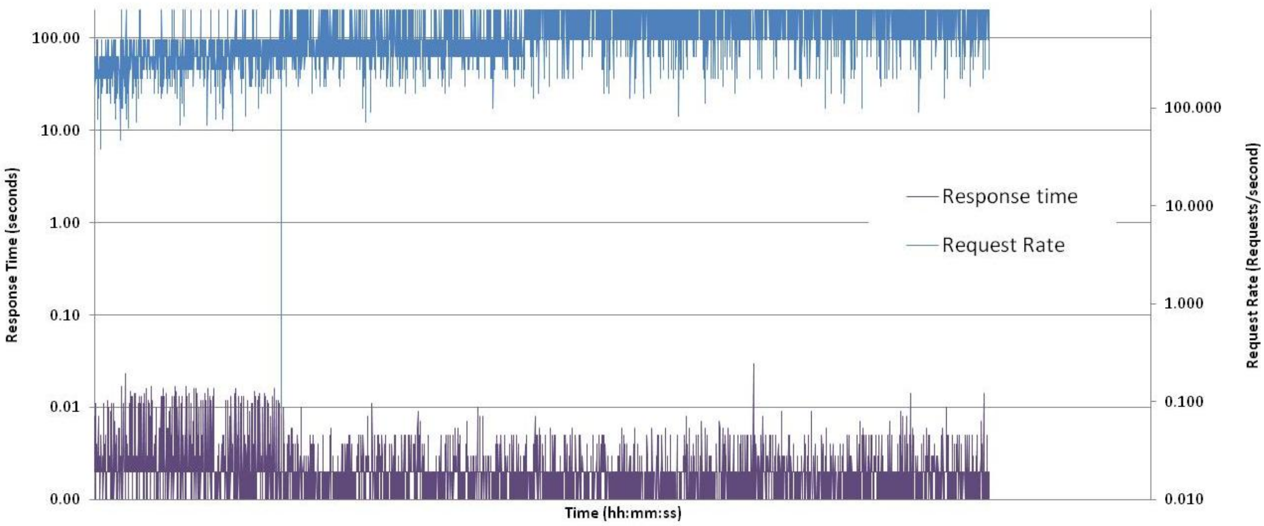


Figure 78 Root CRE Response Time VS CME Request Rate

4.2.1.3 PER-CRE-006- Maximum HTTP Connections

At the local testbed at Primetel we tested for the maximum number of users that can be connected to the CP at the same time. We identified that the limit dependent on the server resources allocated and was not constraint by the CP entity before this limit was reached.

Table 19 shows the results of 5 clients attempting to register content at a single CRE at approximately 0.33 req/sec from each client. It was noticed that certain number of HTTP connection establishment from each client were unsuccessful (88% success rate). This was due to clashing attempts from the different sources in trying to establishing an HTTP connection with the CP. The results are summarised in table form in Table 19.

	CC1	CC2	CC3	CC4	CC5	CRE
Start Time	13:31:56	13:33:28	13:33:39	13:31:42	13:32:40	13:31:42
End Time	15:12:09	15:17:15	15:16:49	15:11:12	15:16:06	15:17:15
Total Run Time	1:40:13	1:43:47	1:43:10	1:39:30	1:43:26	1:45:33
TRT in seconds	6013	6193	6190	5970	6206	6313
Sum Resp. Time	2248.558	2351.951	2332.85	2205.433	2318.852	11457.64
Errors	228	233	244	234	236	1175
Successful Req	1772	1767	1756	1766	1764	8825
Success Rate %	0.886	0.8835	0.878	0.883	0.882	0.8825
Req. Rate per sec.	0.332613	0.322945	0.323102	0.335008	0.322269	1.584033
Response Time (sec)	1.268938	1.331042	1.328502	1.248829	1.314542	0.259674

Table 19 Five Content Clients registering content at CRE

Hence it was noticed that although the max number of HTTP sessions is limited by the server resources at the CRE, the CP software should be improved to handle several requests at the same time more efficiently.

4.2.1.4 PER-CRE-007 - Tests on CRE-CP interface

Initial tests for CRE aimed at obtaining the response time of CRE when attempting to register a particular content. Due to the different stages required in the html form used to publish a content by a user we have broken down the testing into the different stages namely Login, Form 1, Form 2 and Form 3. We also give an estimate of the overall response time required based on uniform traffic load to the CRE.

4.2.1.4.1 Configuration

Keep in mind that config format is YAML.

stress-tester will search for config in /etc/stress-tester.yaml or ./etc/stress-tester.yaml. Last one has more priority.

Example:

```
---
# base part of CP uri
target_base: 'http://10.2.0.7:8090/Publisher/'
# CP login
target_login: 'COMET2'
# CP password
target_password: 'COMET2'
# Log directory (must exist)
log_dir: '/root/test_log'
# Report directory (must exist)
report_dir: '/root/test_report'
# Number of tests per process
reg_num: 480
# Number of processes
proc_num: 1
```

HOW TO USE
1. Configure the test.
2. Start ./test-create.pl and wait until it stop.
3. Start ./test-report.pl and wait until it stop.

4.2.1.4.2 PER-CRE-007a CP Login Response Time

Figure 79 shows the response time required for a CRE to respond to the Login Form published by the CP.

Maximum response time (peak time) obtained was: 5.4223 sec. This happened upon first request and it is not shown on the graph for clarity.

Average Response Time for Login Form: 0.1874 sec,

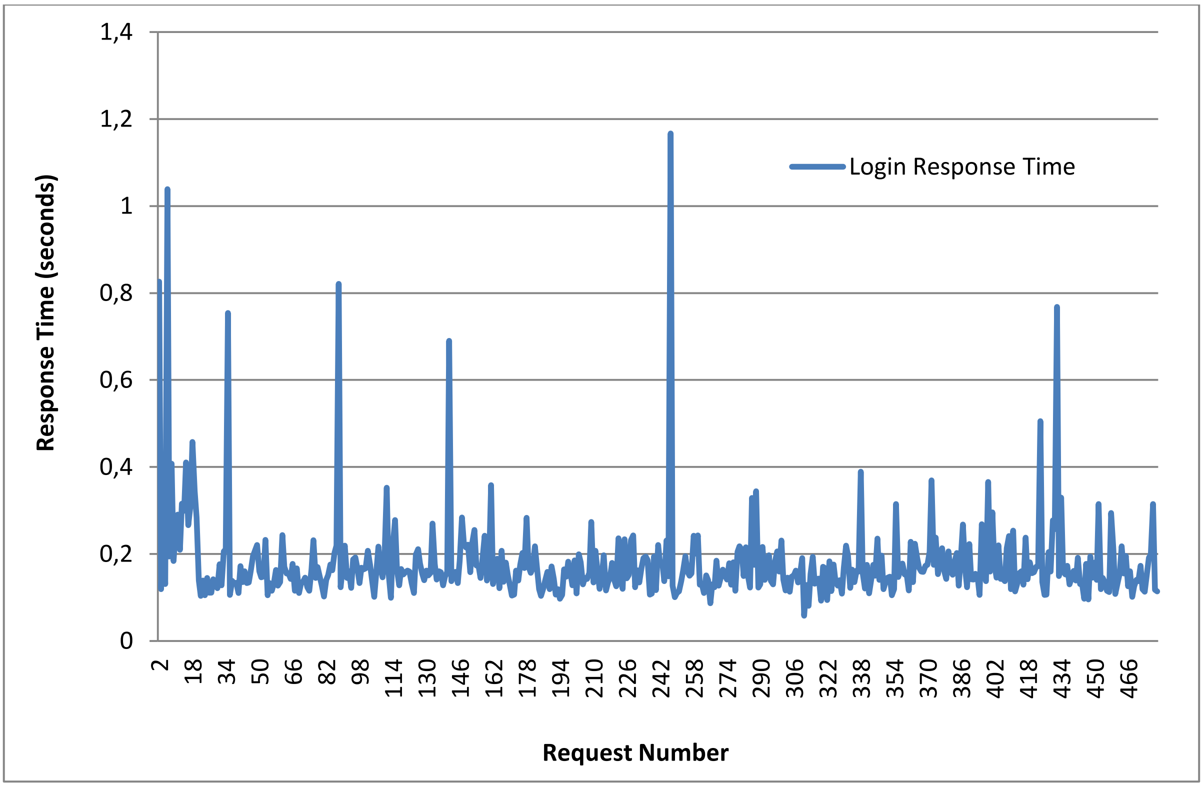


Figure 79 Response time of Content Publication Login Form

4.2.1.4.3 PER-CRE-007b CP Form 1 Response Time

Figure 80 shows the response time required for a CRE to respond to Form 1 published by the CP.

Pick Response Time for Form 1: 1.3875 sec,

Average Response Time for Form 1: 0.0679 sec.

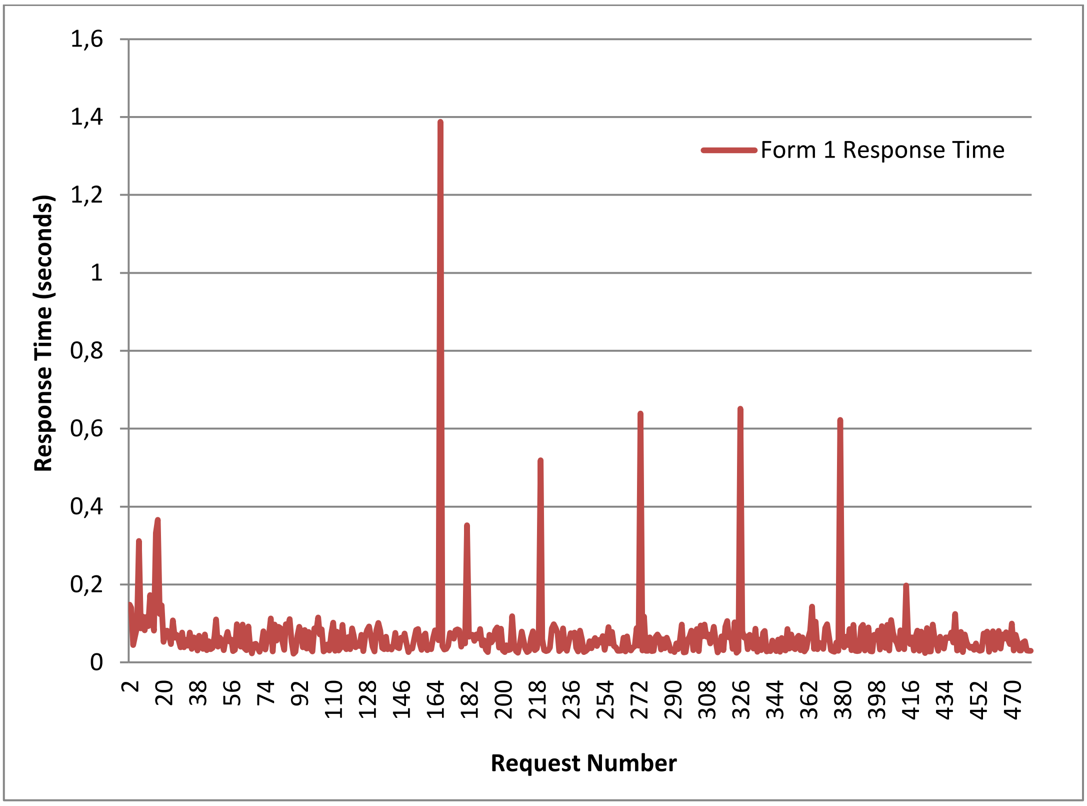


Figure 80 CRE response time to Form 1

4.2.1.4.4 PER-CRE-007c CP Form 2 Response Time

Figure 81 shows the response time required for a CRE to respond to Form 2 published by the CP.

Pick Response Time for Form 2: 0.8592 sec,

Average Response Time for Form 2: 0.0552 sec,

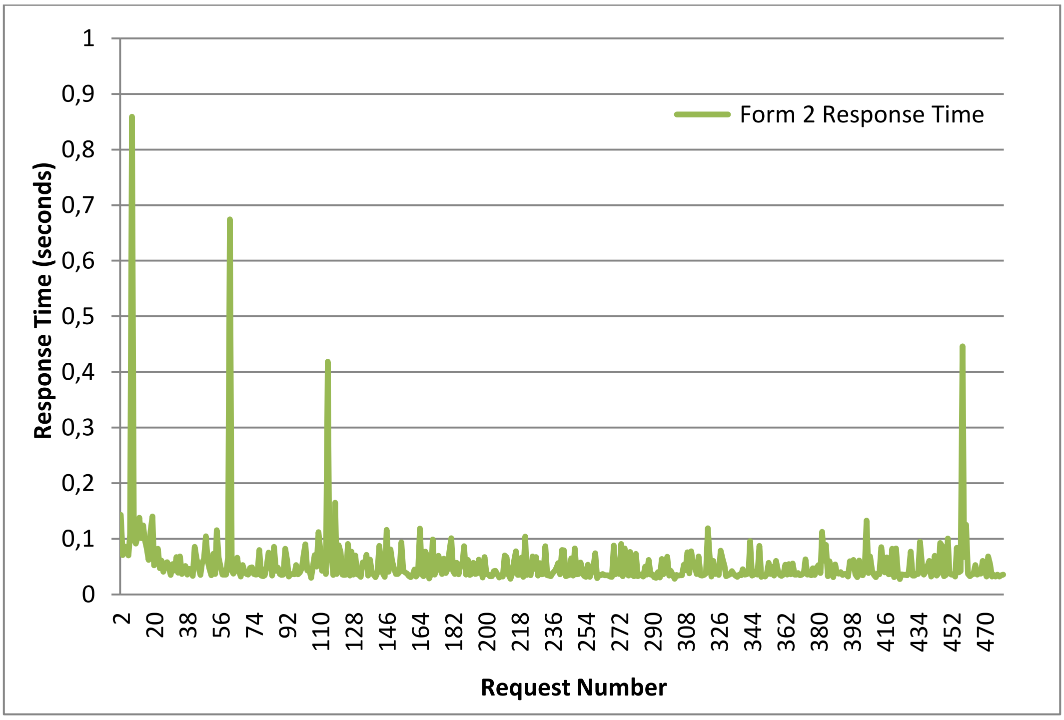


Figure 81 CRE response time to Form 2

4.2.1.4.5 PER-CRE-007d CP Form 3 Response Time

Figure 82 shows the response time required for a CRE to respond to Form 3 published by the CP.

Pick Response Time for Form 3: 2.6291 sec,

Average Response Time for Form 3: 0.1865sec.

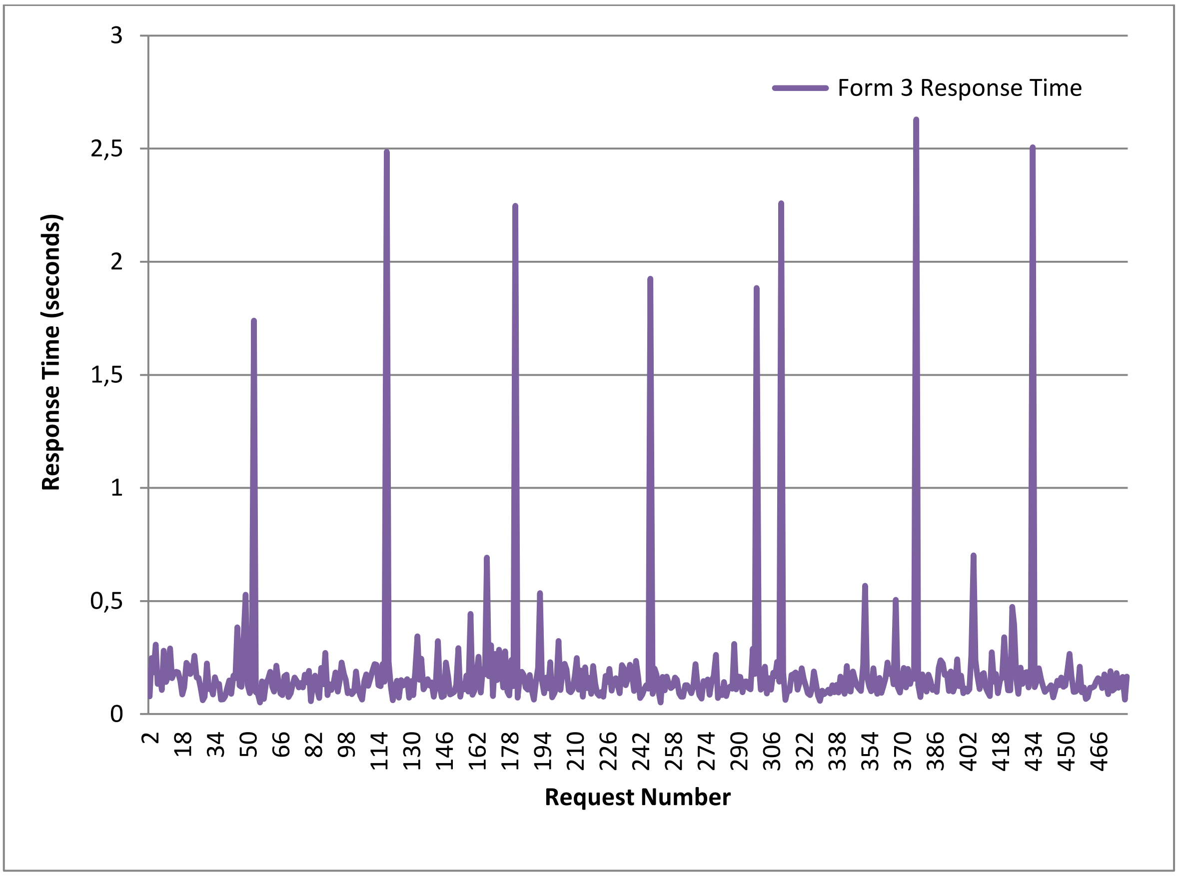


Figure 82 CRE response time to Form 3

4.2.1.4.6 PER-CRE-007e Overall Response Time

Figure 83 shows the response time required for a CRE to respond to all forms published by the CP.

Maximum Response Time 3.2145 sec,

Average Response Time: 0.4979 sec.

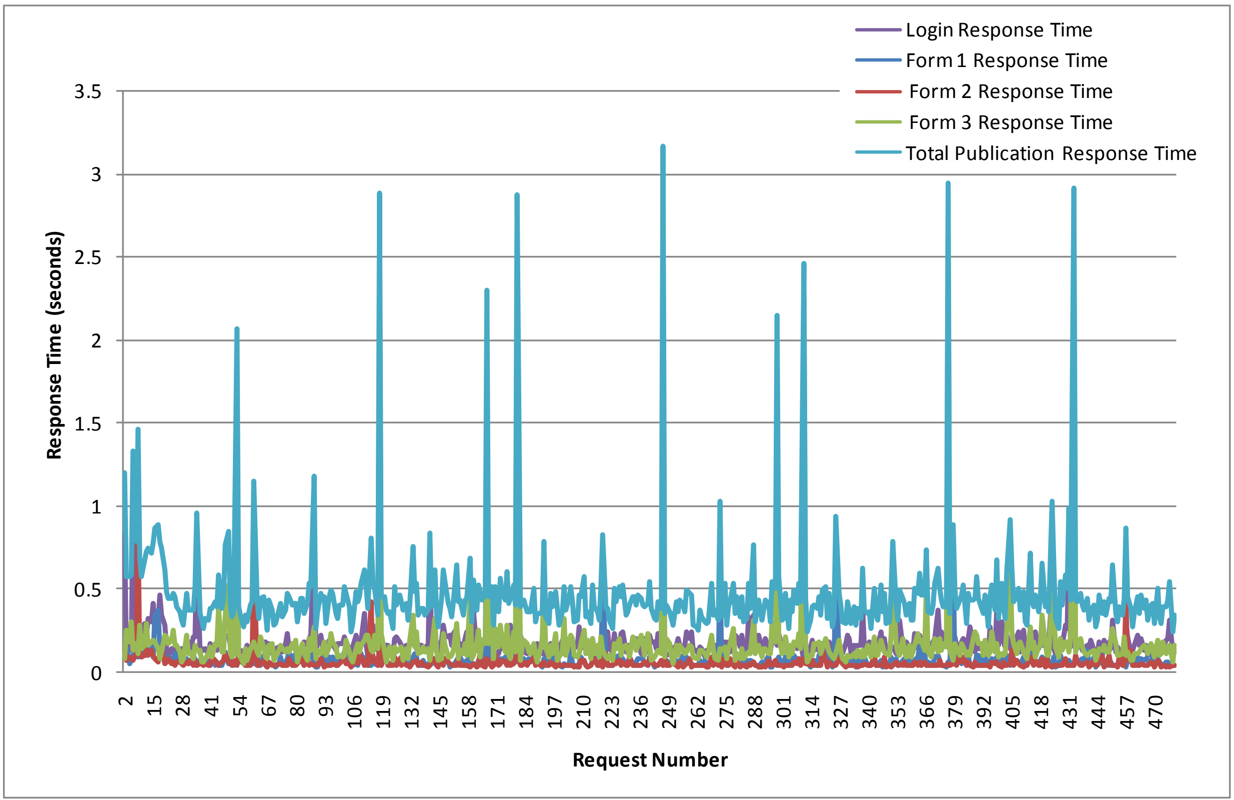


Figure 83 CRE total response time

4.2.1.4.7 PER-CRE-007f Overall Response Time with Errors

For the overall response time we repeated the CRE response test for the same set of CP requests but this time with the addition of dummy errors testing hence obtaining the impact on future requests. It is shown that in the case of problematic requests the CRE requests to follow could delay up to 6.18 seconds to respond.

Here is more detailed summary of the collected results:

Peak CRE response time for Login Form: 1.0569 sec,

Peak CRE response time for Form 1: 6.0887 sec,

Peak CRE response time for Form 2: 0.6518 sec,

Peak CRE response time for Form 3: 3.2934 sec,

Average CRE response time for Login Form: 0.1968 sec,

Average CRE response time for Form 1: 0.0819 sec,

Average CRE response time for Form 2: 0.0528 sec,

Average CRE response time for Form 3: 0.2232 sec,

Maximum Overall: 6.18885397911072 sec,

Forced Errors: 185; Successful requests: 296; Total: 481;

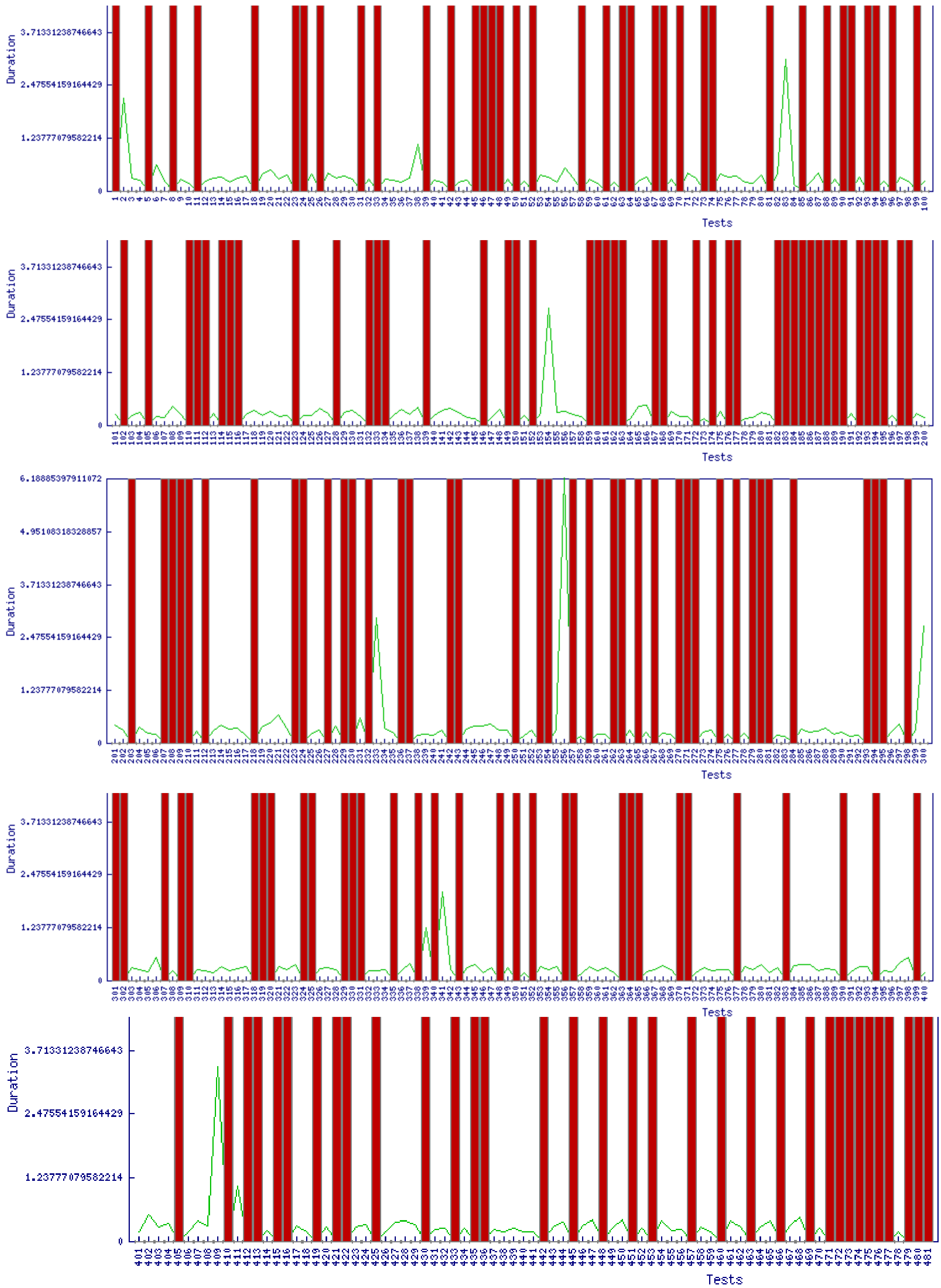


Figure 84 Overall Response Time with Errors

4.2.2 CME Tests

4.2.2.1 PER-CME-001 CC-CME interface tests

The validation tests below relate to the Content Client (CC) – Content Mediation Entity (CME) stress testing. For this validation test a CC stress tester has been developed which sends CC requests to the CME at a Poisson Rate with a configurable average. Figure 85 shows the request rates set on the client side in requests per second, shown in red. In blue we show the arrival rate of these packets to the CME showing the impact of processing, transmission and network interference.

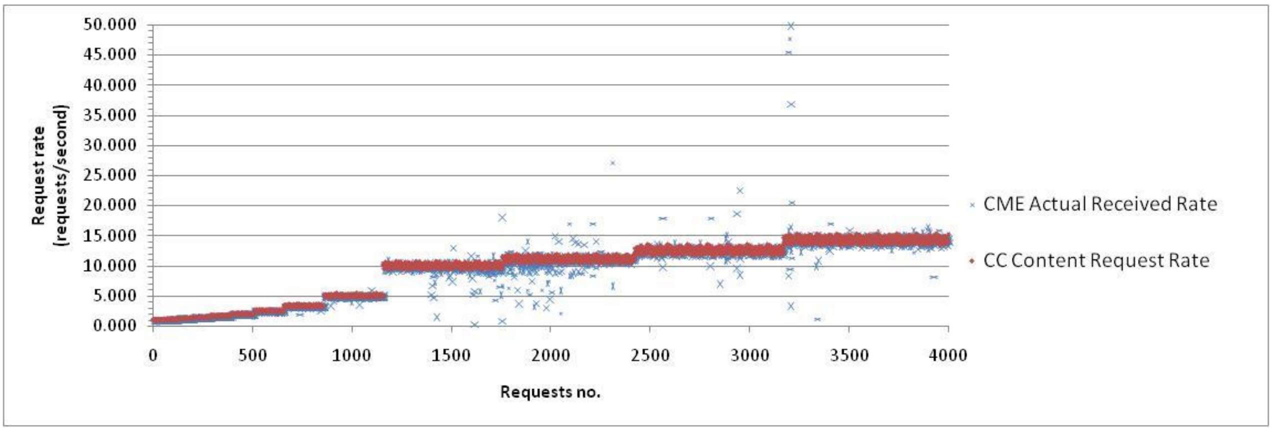


Figure 85 CC Poisson request rate vs Actual arrival rate at CME

Figure 86 shows the CME CPU (in purple) and CME Memory consumption (in green) against the CC request rate. The results show that after about 10 requests per second towards the same CME begins to consume most of the CPU. This is because the CME as it is presently developed, processes a single request at a time in the order of 200ms with future requests being queued. Hence about 5 requests per second can be more or less processed effectively, with any higher rates resulting to undesired high CPU consumption. Since the requests are fairly small in the order of kilobytes.

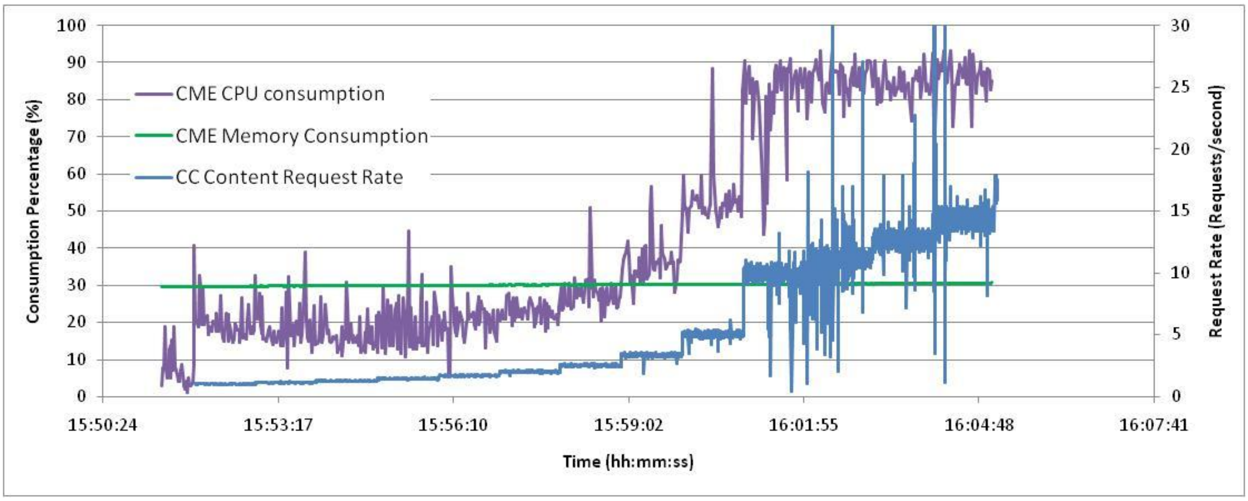


Figure 86 CPU and Memory Consumption versus Request Rate

Figure 87 shows the CME response time against the CC request rate. The graph shows that when the CC request rate is increased to 10 requests per second the CME response time suddenly increases sharply to more than 10 seconds, hence an unsatisfactory response time.

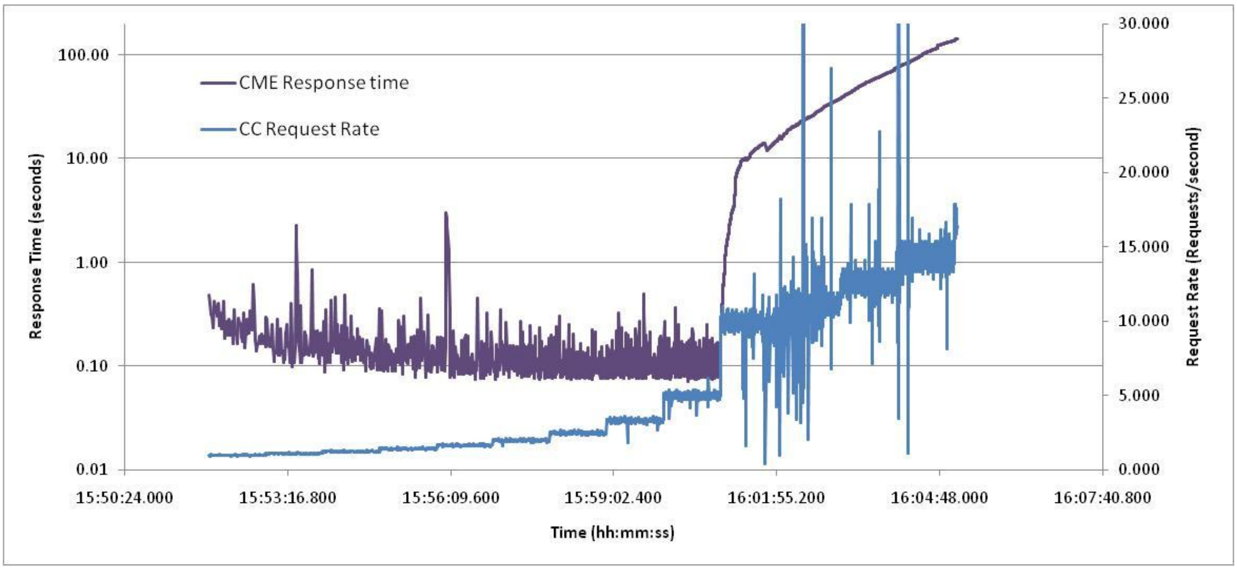


Figure 87 CME Response Time versus CC Request Rate

4.2.2.2 PER-CME-002 Testing the RAE-CME interface

Figure 88 and Figure 89 shows the CPU and Memory at the CME with average query rates at about 500 queries/second and 600 queries/second. It has to be noted that this “maximum” limits are influenced by the TCP protocol established between the RAE and CME entities. Even when the query rate is increased in the order of 1K, 10K and 100K queries per second the TCP protocol constraints the rate to ensure 100% success.

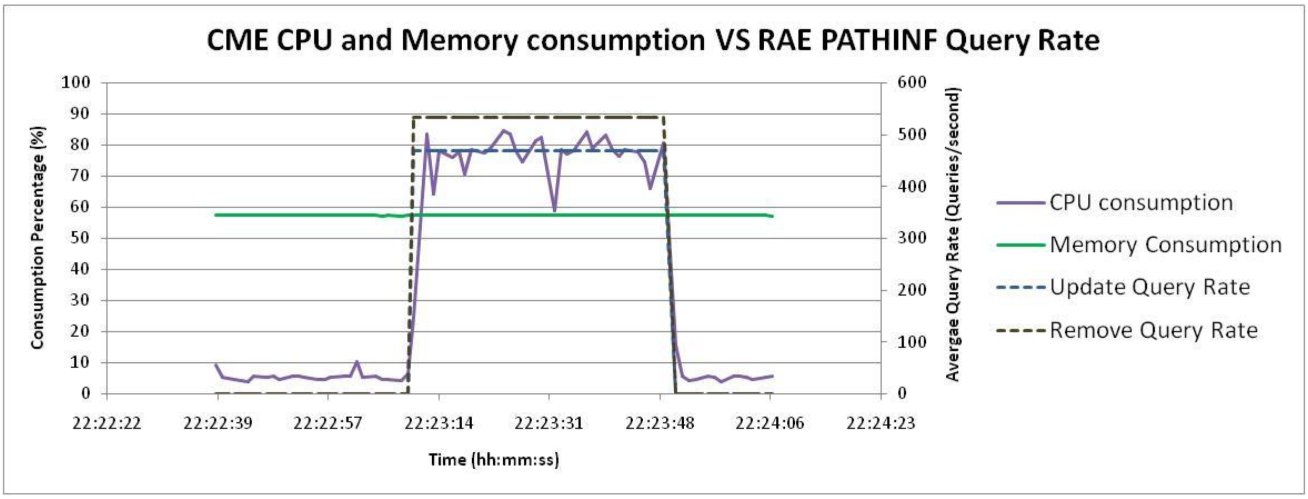


Figure 88 CME CPU and Memory consumption at near 500 queries/sec from RAE

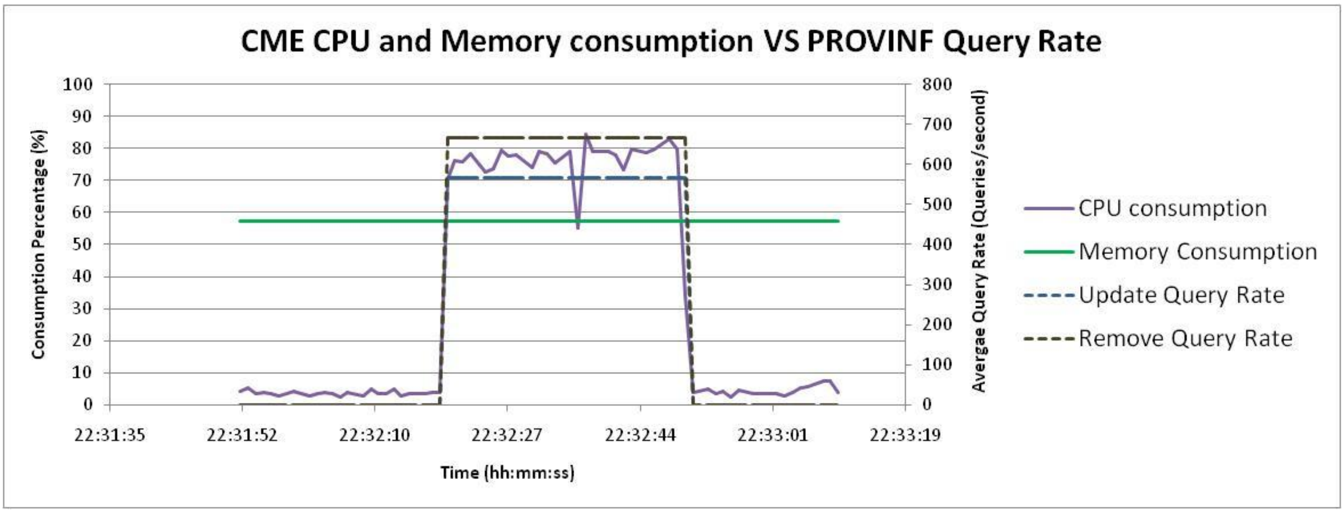


Figure 89 CME CPU and Memory consumption at near 600 queries/sec from RAE

4.2.3 SNME Tests

The whole of the SNME tests, from PER-SNME-001 to PER-SNME-009, had been performed outside of federated testbed in a separate environment. Features of the machines involved in the test are the following:

- SNME: OS: Ubuntu 10.04; CPU: 2.40GHz; RAM: 3.20GB
- Stress Tester (SMA simulator): OS: Windows 7 (64bits); CPU: 2.40GHz; RAM: 4GB

4.2.3.1 Tests on SMA-SNME

4.2.3.1.1 PER-SNME-001 and 002 Tests on CS-SNME interface

The objective of the test is to characterize the behavior of SNME CPU and Memory as a function of the number of connected content servers in order to get the recommended number of servers that can be managed by a SIC in the SNME.

The operation for the test is:

- Run SNME in order to enable SIC.
- Launch SMA simulator sending n messages with server status to SIC.
- Get information from the behavior of SNME CPU and Memory.
- Repeat the test increasing gradually the number of messages sent to SIC.

According the results and considering CPU load as the determining factor, the recommended number of servers managed by a SIC has been set to 500.

Figure 90 PER-SNME-001/002

4.2.3.1.2 PER-SNME-003 SNME-SYN Test

The objective of the test is to characterize the behavior of SNME as percentage of SYN to SERVER STATUS vs. the number of connected CSs in order to get the recommended number of servers that can be managed by a SIC in the SNME.

The operation for the test is:

- Run SNME in order to enable SIC.
- Launch SMA simulator sending n messages with server status to SIC.
- Get information from the percentage of SYN messages.
- Repeat the test increasing gradually the number of messages sent to SIC.

According the results of test PER-SNME-001 and PER-SNME-002 and considering CPU load as the determining factor, the recommended number of servers managed by a SIC has been set to 500. In this test the SYN percentage is not the decisive factor but when the server load is above 300 servers the percentage grows close to 50% so the limit of 500 servers can be set as the number of recommended servers managed by SIC.

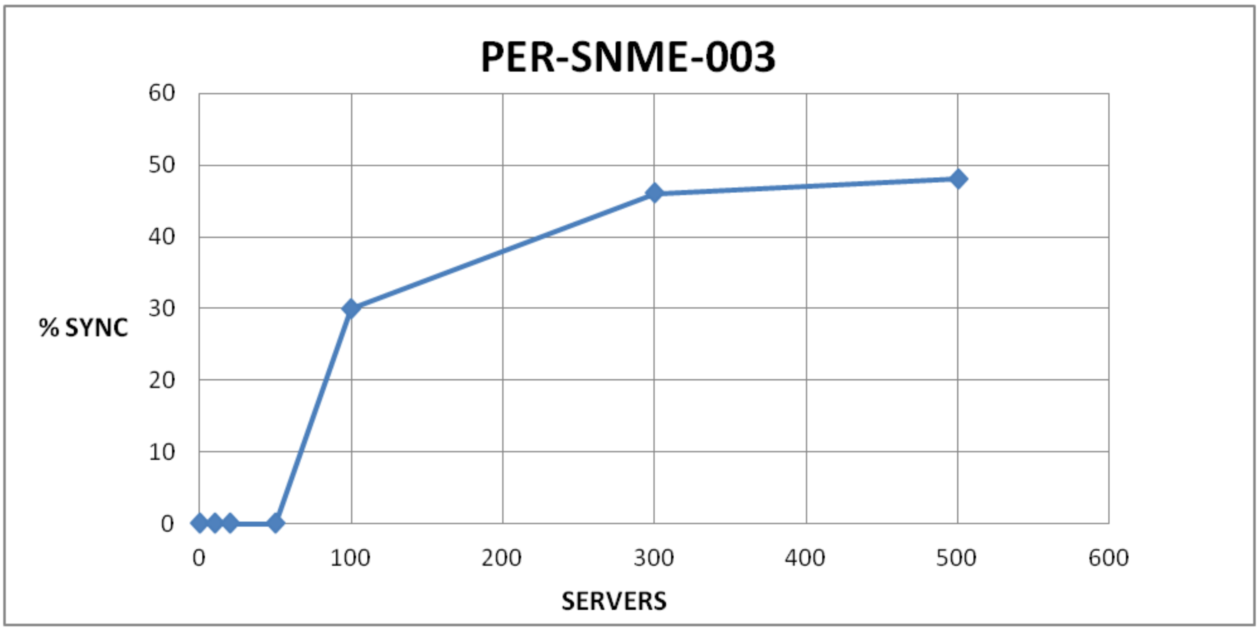


Figure 91 PER-SNME-003

4.2.3.2 Tests on CME-SNME Interface

4.2.3.2.1 PER-SNME-004

The objective of the test is to characterize the behavior of SNME CPU as a function of the number of queries per second SNME (SAS) receives from CME in order to get the optimal range as a function of query rates for different CS population. CS limit of 500 servers established in PER-SNME-001 to PER-SNME-003 is used for emulating a low CS population (20%, 100 servers), medium (70%, 350 servers) and high (90%, 450 servers).

The operation for the test is:

- Run SNME in order to enable SIC and SAS.
- Launch SMA simulator sending messages with server status to SIC.
- Launch CME simulator sending queries asking for a number of servers to SAS. The number of servers remains constant.
- Get information from the behavior of SNME CPU
- Repeat the test increasing gradually the number of queries sent to SAS.
- Repeat the test for different CS population: low, medium and high.

According the results and considering CPU load as the determining factor, the recommended number of servers managed by SAS has been set to 500. Within this limit and a CPU load lower to 80%, it can be able to state that system is still inside the safe side.

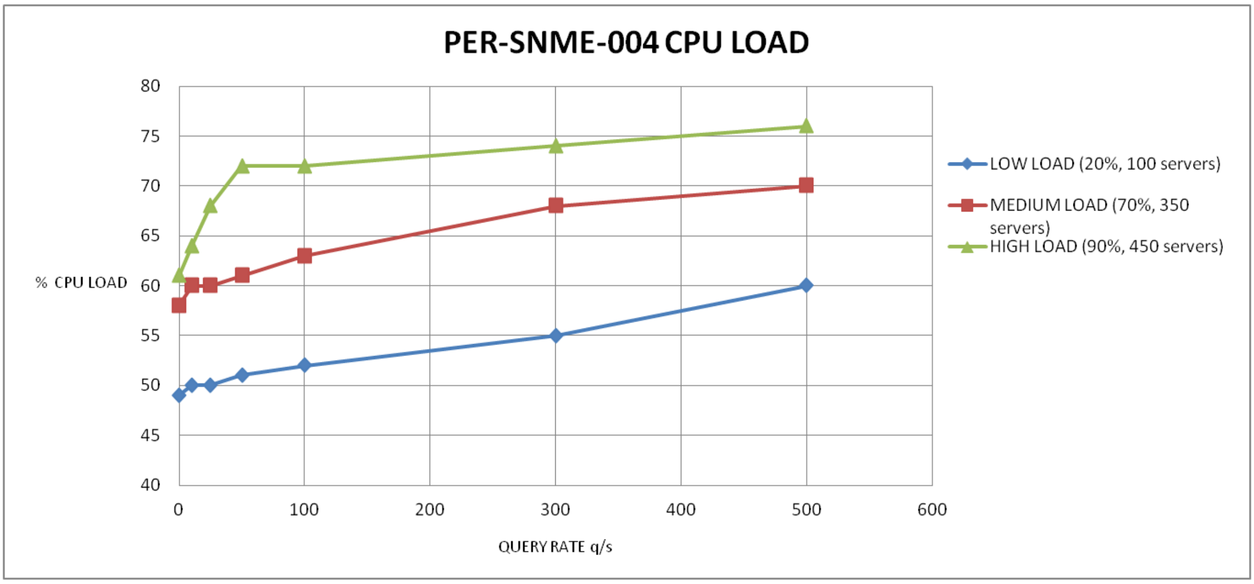


Figure 92 PER-SNME-004

4.2.3.2.2 PER-SNME-005

The objective of the test is to characterize the behavior of SNME CPU as function of amount of CS in queries SNME receives from CME in order to get the optimal range as a function of amount of CS in query and query rates for CS population.

In this test it is only emulated a high CS population (90%, 450 servers) from the limit established in PER-SNME-001 to PER-SNME-003 (500 servers). Only high population has been consider for the test due to in the last case it was determinate that despite the high load, the process is operating in a safe range.

The simulation of query rates use the three highest rates established in PER-SNME-004: 100, 300 and 500 queries per second.

The operation for the test is:

- Run SNME in order to enable SIC and SAS.
- Launch SMA simulator sending messages with server status to SIC emulating a high CS population (90%).
- Launch CME simulator sending queries asking for a number of servers to SAS.
- Get information from the behavior of SNME CPU.
- Increase gradually the number of servers included in the queries.
- Repeat the test for query rates established in PER-SNME-004.

According the results, considering CPU load as the determining factor and emulating the highest query rate (500 queries per second) the recommended number of servers include in a query tan can be managed by SAS has been set to 300. Upper this limit SNME CPU reaches values close to 80% and higher.

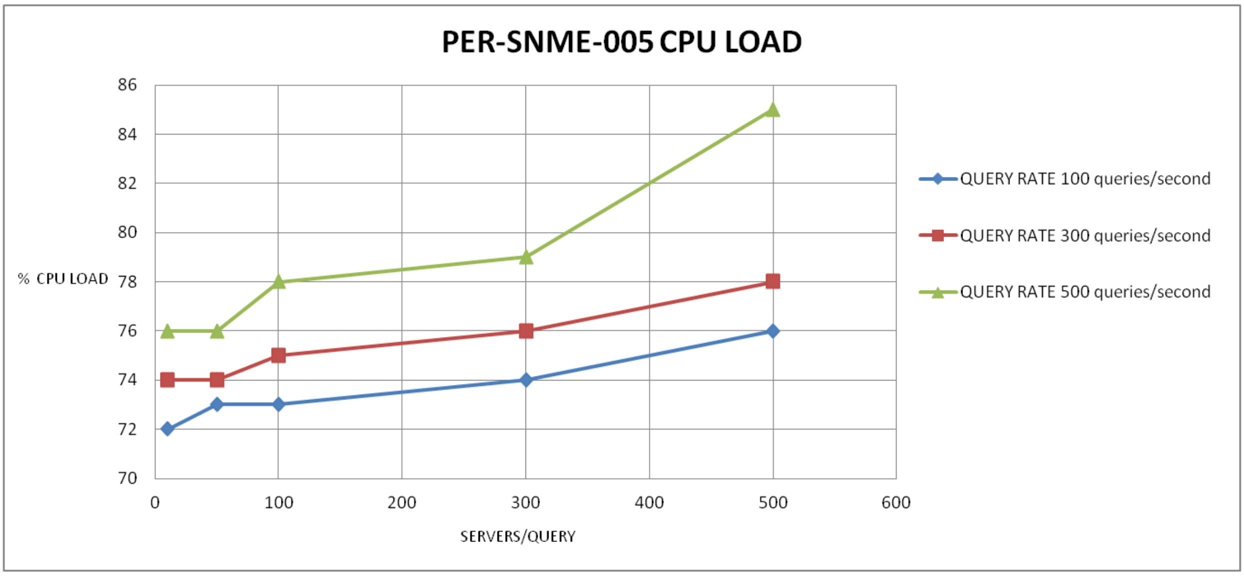


Figure 93 PER-SNME-005

4.2.3.2.3 PER-SNME-006

The objective of the test is to characterize the behavior of SNME memory as a function of the number of queries per second SNME (SAS) receives from CME in order to get the optimal range as a function of query rates for different CS population. CS limit (500 servers) established in PER-SNME-001 to PER-SNME-003 is used for emulating a low CS population (20%, 100 servers), medium (70%, 350 servers) and high (90%, 450 servers).

The operation for the test is:

- Run SNME in order to enable SIC and SAS.
- Launch SMA simulator sending messages with server status to SIC.
- Launch CME simulator sending queries asking for a number of servers to SAS. The number of servers remains constant.
- Get information from the behavior of SNME memory
- Repeat the test increasing gradually the number of queries sent to SAS.
- Repeat the test for different CS population: low, medium and high.

According the results, memory is not the decisive factor to set the recommended number of servers managed by SAS. In the same way that PER-SNME-004, memory load at highest values are within a safe range, under 50%, so also considering CPU the decisive factor, the number of recommended server is set by the results of PER-SNME-004 test, that is 500.

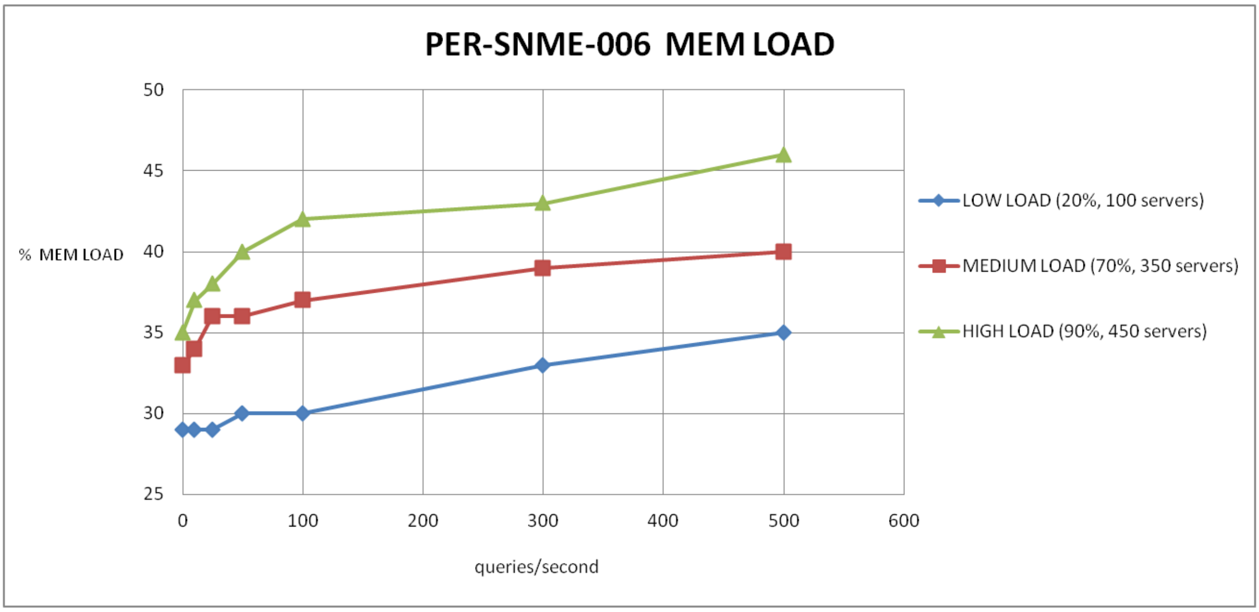


Figure 94 PER-SNME-006

4.2.3.2.4 PER-SNME-007

The objective of the test is to characterize the behavior of SNME memory as a function of amount of CS in queries SNME receives from CME in order to get the optimal range as a function of amount of CS in query and query rates for CS population.

In this test it is only emulated a high CS population (90%, 450 servers) from the limit established in PER-SNME-001 to PER-SNME-003 (500 servers). Only high population has been consider for the test due to in the last case it was determinate that despite the high load, the process is operating in a safe range.

The simulation of query rates use the three highest rates established in PER-SNME-004: 100, 300 and 500 queries per second (Figure 95).

The operations for the test are:

- Run SNME in order to enable SIC and SAS.
- Launch SMA simulator sending messages with server status to SIC emulating a high CS population (90%).
- Launch CME simulator sending queries asking for a number of servers to SAS.
- Get information from the behavior of SNME CPU.
- Increase gradually the number of servers included in the queries.
- Repeat the test for query rates established in PER-SNME-006.

According the results, memory is not the determining factor so the optimal query rate that can be managed by SAS is set by the results of CPU load at PER-SNME-005. Memory load for the highest values are lower 55% so it can be consider them within a safe range.

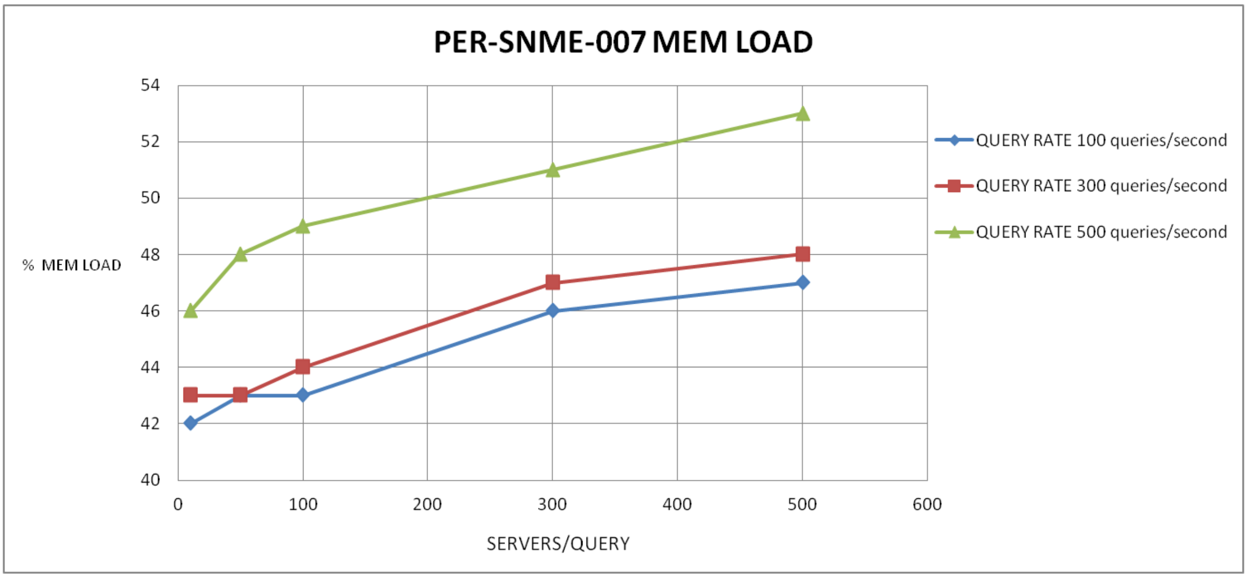


Figure 95 PER-SNME-007

4.2.3.2.5 PER-SNME-008

The objective of the test is to characterize the Response Time of SNME as a function of query rate in order to get SNME Response Time (Minimum, Maximum, Mean and 95 percentile) vs. Query Rates according to the selected Poisson Distributions for each Server Population. CS limit established in PER-SNME-001 to PER-SNME-003 (500 servers) is used for emulating a low CS population (20%, 100 servers), medium (70%, 350 servers) and high (90%, 450 servers) (see Figure 96).

Poisson distribution is used in order to simulate different query rates. The formula for query rate (λ) can be described as: 1 / ((time)) / (number of queries).

The operation for the test is:

- Run SNME in order to enable SIC and SAS.
- Launch SMA simulator sending messages with server status to SIC.
- Launch CME simulator sending queries following a Poisson distribution asking for a number of servers to SAS. The queries includes in a query remains constant.
- Calculate the response time from SNME.
- Repeat the test for different CS population: low, medium and high.
- Repeat the test for different Poisson distribution patterns.

According the results and considering as valid response time any record lower to 1 second (1000 ms), an optimal query rates can be set to λ =1.66, this mean 500 queries distributed along 300 seconds and λ =3.33, this mean 1000 queries distributed along 300 seconds.

Minimum values are not included in the chart due to they can be regarded as negligible.

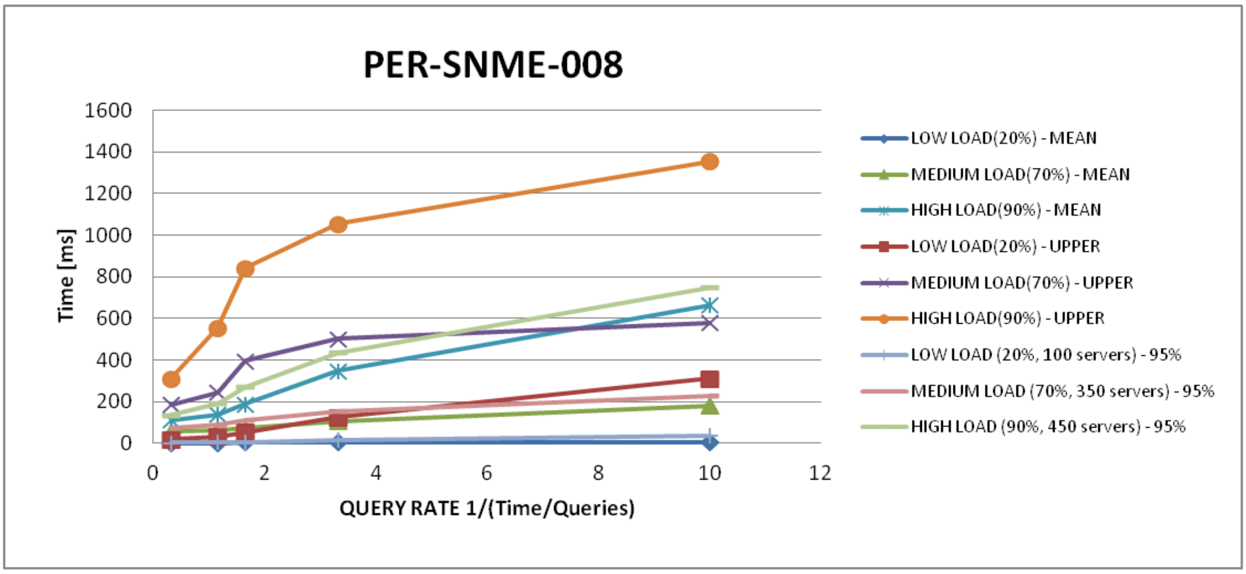


Figure 96 PER-SNME-008

4.2.3.2.6 PER-SNME-009

The objective of the test is to characterize the Response Time of SNME as a function of the size of queries SNME receives from CME and query rate, in order to get SNME Response Time (Minimum, Maximum, Mean and 95 percentile) vs. Query Rates according to different amounts of CS IDs in query, for each of the couples Medium Traffic/Medium Occupation, Medium/High, High/Medium,High/High as defined by the results of previous tests. CS limit established in PER-SNME-001 to PER-SNME-003 (500 servers) is used for emulating medium population (70%, 350 servers) and high (90%, 450 servers) (see Figure 97 and Figure 98).

Medium and high Query rates established from PER-SNME-008 are λ =1.66 and λ =3.33.

The operation for the test is:

- Run SNME in order to enable SIC and SAS.
- Launch SMA simulator sending messages with server status to SIC.
- Launch CME simulator sending queries following the different patterns of couples Traffic / Occupation.
- Calculate the response time from SNME.
- Repeat the test increasing gradually the number of servers included in the queries.

According the results and considering as valid response time any record lower to 1 second (1000 ms), an optimal number of servers per query can be set to 100.

Minimum values are not included in the chart due to they can be regarded as negligible.

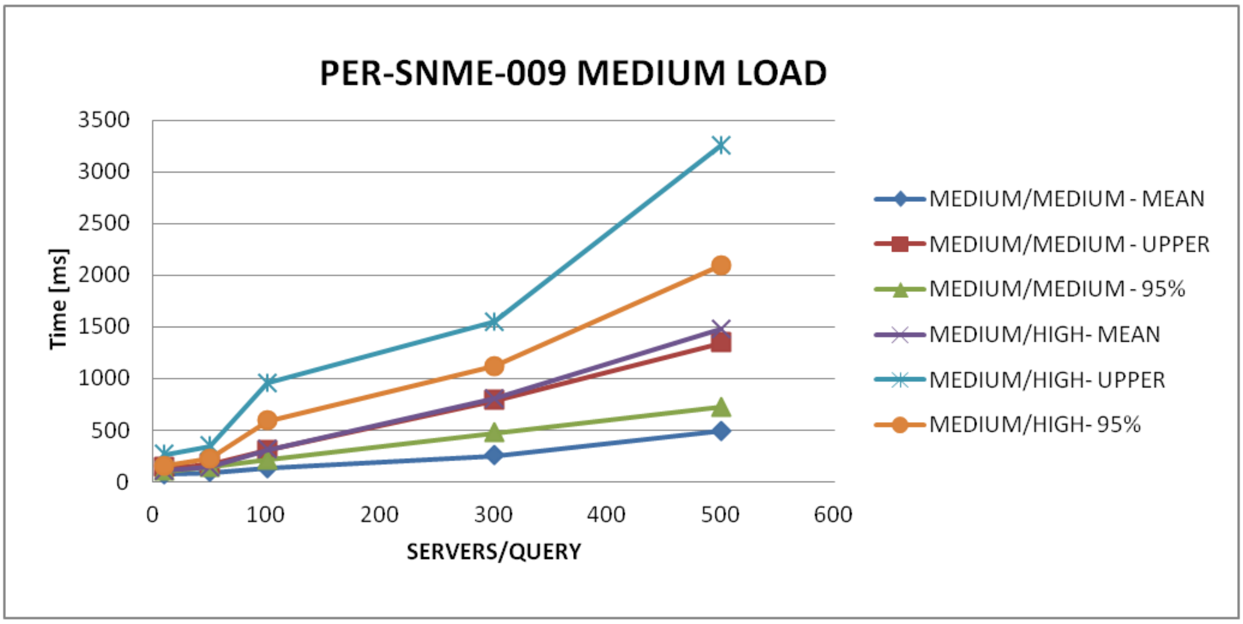


Figure 97 PER-SNME-009

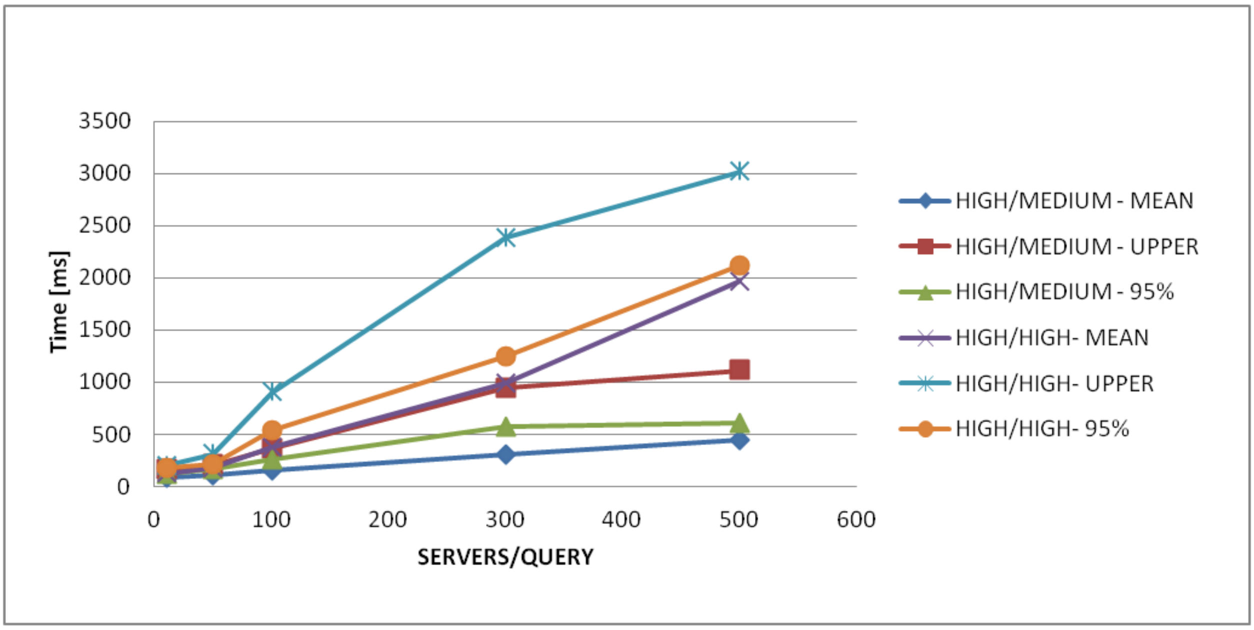


Figure 98 PER-SNME-009

4.2.4 RAE Tests

The objective of RAE tests is to evaluate the performance of RAE prototype. The tests focused on two basic features: (1) ability to handle large number of prefixes and (2) the evaluation of routing convergence time in the federated testbed.

4.2.4.1 Ability to handle large number of prefixes

The PER-RAE-001 test evaluates the ability of RAE to store and process large number of prefixes. Let us recall that RAE was designed for performing multi-criteria and multi-path inter-domain routing in the Internet scale network. Therefore, we assume that RAE should be able to handle similar number of prefixes as currently process by routers running in the Tier 1 domains. Based on the analysis of datasets provided by CAIDA [9],[10], [11], we identified that current BGP routers handle around 300 000 prefixes. Therefore, in this test we assume that RAE should handle at least 300 000 prefixes.

Taking into account that federated testbed is deployed on virtual machines that have limited processing power and memory, the PER-RAE-001 test was performed in the standalone testbed presented in the Figure 99.

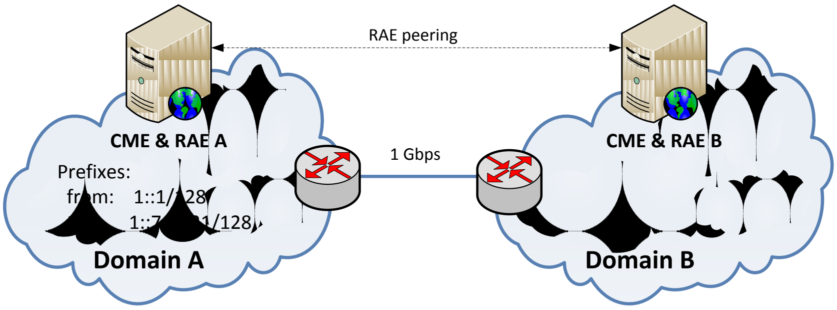


Figure 99: The standalone testbed used for PER-RAE-001 test

This testbed models two peering Tier 1 domains with deployed RAE entities. They exchange information about available prefixes and calculate routing paths. Both RAEs run on the ordinary PC machine with Intel(R) Core(TM)2 Duo CPU E8500 @ 3.16GHz processor and 16 GB RAM. In our experiment, we assume that domain A advertises a number of prefixes, starting from 1 prefix (1::1/128) up to 500 0000 prefixes (1::7:A121/128). Once, the routing becomes stable, we break connectivity between domains to assess RAE performance for prefix withdrawal. Each experiment was repeated 100 times to get credible results. In each test case, we measure the following metrics:

- **Routing Convergence Time (RCT)**, defined as the total amount of time that elapses between the occurrence of stressing event and the time instant when the last update message caused by this event has been processed by RAE to reach stable routing.
- **The memory used by RAE to store prefixes**. We monitor both the physical memory and total amount of memory (physical, swap)
- **The CPU used by RAE to process messages**. The CPU was measured in 1 sec intervals during the convergence process.

The results obtained for prefix advertisement and withdrawal are presented in Table 20 and Table 21, respectively. We can observe that:

- In case of prefixes advertisement, the RCT, the used memory and CPU load is linearly proportional to the number of prefixes. The lowest RCT value is about 5s, which conforms to the inter-message timeout assumed for communication between RAE and CME. The RCT for the largest number of prefixes is about 3 min and occupied memory is about 400 MB. Moreover, we can observe that starting from 100 000 prefixes, the RAE consumes almost 100% CPU power leading to significant increase of convergence time. We can conclude that the CPU was the main bottleneck in this experiment.

- In case of prefixes withdrawal, the RCT, the used memory and CPU load is linearly proportional to the number of prefixes. The lowest RCT value is below 1 ms because RAE immediately informs CME about prefix withdrawals (in this case the inter-message timeout is not applied). The RCT for the largest number of prefixes is about 1 min. This value is 3 times lower than delay measured for prefix advertisement, because processing is less complicated. Moreover, we can observe that RAE releases memory used to store prefixes. Similar to the prefix advertisement experiment, we observe that CPU became fully occupied in case of 100 000 or more prefixes. Again, we identify CPU power as the main limitation in this experiment.

No prefixes	**Mean Routing Convergence Time (RCT) [s]**	**Used memory total /physical [M Bytes]**	**Mean CPU load [%]**
1	5.000	9.700 / 4.432	1%
100	5.029	9.700/ 4.508	2%
1 000	5.197	10.476 / 5.212	19%
10 000	6.836	17.496 / 12.196	45%
100 000	37.436	88.512 / 83.268	96%
300 000	124.000	242.624 / 237.396	98%
500 000	185.818	400.652 / 395.380	98%

Table 20: Results of PER-RAE-001 test for prefix advertisement

No prefixes	**Mean Routing Convergence Time (RCT) [s]**	**Used memory total /physical [M Bytes]**	**Mean CPU load [%]**
1	<0.001*	9.700 / 4.372	1%
100	0.022	9.700 / 4.372	2%
1 000	0.128	9.700 / 4.372	13%
10 000	1.031	9.700 / 4.372	51%
100 000	12.797	9.700 / 4.372	91%
300 000	38.645	9.700 / 4.372	98%
500 000	64.173	9.700 / 4.372	98%

* RCT was below resolution of measurement tool (lower than 1 ms)

Table 21: Results of PER-RAE-001 test for prefix withdrawal

The performed tests confirmed that RAE prototype is able to handle large number of prefixes. In analysed cases, the RAE running on the ordinary PC machine was able to handle the same number of prefixes as currently available in the Internet. The routing convergence time for prefix advertisement was about 3 min due to CPU overload. Anyway, the obtained RCT values are at a similar level as reported in [12] for the inter-domain routing. During the experiment, the CPU was recognised as the main bottleneck. Therefore, the RAE prototype should be optimised from the CPU usage point of view. Moreover, we recommend deploying RAE on machines with powerful processor to reduce the RCT.

4.2.4.2 Routing convergence time

The PER-RAE-002 and 003 tests evaluate the Routing Convergence Time (RCT) in the federated testbed controlled by RAE. The RCT is one of the basic performance indicators related to routing protocols [12], which assess how routing protocol reacts on the stressing events, e.g. advertisement/withdrawal of prefixes, link failures, etc. Following requirements presented in D5.1 [5], we defined RCT as the total amount of time that elapses between the occurrence of stressing event and the time instant when the last update message caused by this event has been processed to reach stable routing. In our experiments we consider three stressing events that are:

- **Prefix advertisement** in stub domain WUT 64. This event corresponds to the situation when new network or domain becomes available. The RAE propagates new prefix to other domains and calculates routing paths.
- **Prefix withdrawal** in stub domain WUT 64. This event corresponds to the situation when the previously available network or domain becomes unavailable. In this case, the RAE removes information about unavailable prefix.
- **Link failure/repair** between transit domains. Once link failure is detected, the RAE recalculates routing paths to avoid damaged link and switches affected paths to alternative paths.

The PER-RAE-002 and 003 tests were performed in the federated testbed, which was extended by stub domain WUT 64 connected to domain WUT 63. The topology of extended federated testbed is presented in Figure 100.

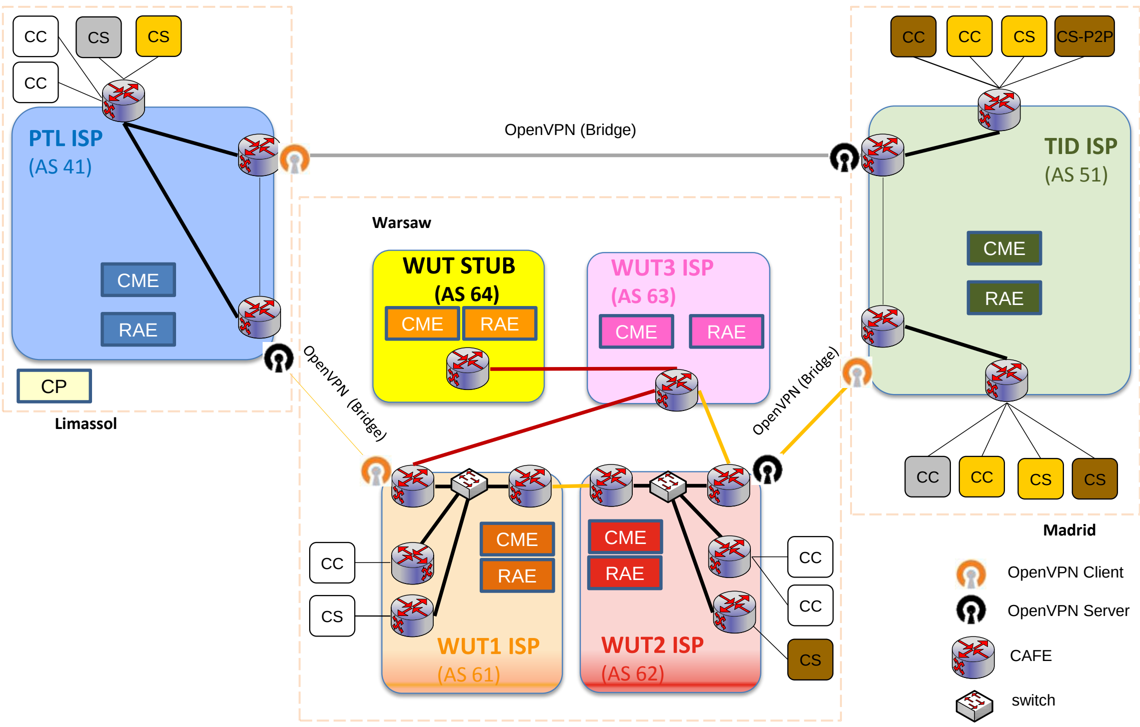


Figure 100 Topology of the extended federated testbed used for PER-RAE-002-003 tests

In PER-RAE-002 test, we advertise prefix 2001:67c:24cc:31f1::/64 in the stub domain WUT 64 and measure the RCT based on the log files collected from RAEs located in all domains. Once the routing becomes stable, we check whether RAE propagates information about new prefix and correctly calculates routing paths towards this prefix. As the next step, we withdraw the prefix from the stub domain and again we measure the RCT for prefix withdrawal and check RAE routing table. The presented procedure was repeated about 150 times to get credible results. We assume 5 min interval between consecutive prefix advertisement and withdrawal events to reach stable routing in the federated testbed.

Figure 101 presents the exemplary RAE log file collected from all RAEs for prefix advertisement. It illustrates how RAEs build routing paths after prefix advertisement in the stub domain WUT 64. At the beginning, we can see that stub domain WUT 64 propagate prefix 2001:67c:24cc:31f1::/64 to peering domain 63. Next, this prefix is propagated to domains 62 and 61. During this process alternative paths are discovered and propagated. Finally, we can see the set of alternative paths {64,63,61}, {64,63,62,61} and {64,63,62,51,41,61}.

```
2012-12-01 21:00:18,737 DEBUG [comet.rae.tcp_connection] '[2001:67c:24cc:31e1::ae:4]:10001' sending message
2012-12-01 21:00:18,747 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #63
2012-12-01 21:00:18,748 DEBUG [comet.rae.rae_logic] updating prefix 2001:67c:24cc:31f1::/64 in class #1
2012-12-01 21:00:18,749 DEBUG [comet.rae.rae_logic] best paths: length=2 {64,63} [d=0.006,l=6.99999e-06,b=1e+07]
2012-12-01 21:00:19,134 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #62
2012-12-01 21:00:19,134 DEBUG [comet.rae.rae_logic] updating prefix 2001:67c:24cc:31f1::/64 in class #1
2012-12-01 21:00:19,145 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,62} [d=0.0165,l=1.59999e-05,b=1e+07]
2012-12-01 21:00:19,213 DEBUG [comet.rae.session] sending KEEPALIVE to [#61]
2012-12-01 21:00:19,213 DEBUG [comet.rae.tcp_connection] '[2001:67c:24cc:31c1::ae:1]:54489' sending message
2012-12-01 21:00:19,298 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #61
2012-12-01 21:00:19,298 DEBUG [comet.rae.rae_logic] updating prefix 2001:67c:24cc:31f1::/64 in class #1
2012-12-01 21:00:19,298 DEBUG [comet.rae.tcp_connection] '[2001:67c:24cc:31e1::ae:3]:10001' received message
2012-12-01 21:00:19,302 DEBUG [comet.rae.tcp_connection] '[2001:67c:24cc:31d1::ae:2]:59994' sending message
2012-12-01 21:00:19,303 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,61} [d=0.0165,l=1.59999e-05,b=1e+07]
2012-12-01 21:00:19,310 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #51
2012-12-01 21:00:19,311 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,62} [d=0.0165,l=1.59999e-05,b=1e+07]
2012-12-01 21:00:19,311 DEBUG [comet.rae.tcp_connection] '[2a02:9008:0:1911:0:50:56a3:51]:56413' sending message
2012-12-01 21:00:19,323 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #63
2012-12-01 21:00:19,324 DEBUG [comet.rae.cme_connection] '[2001:67c:24cc:31d1::ae:2]:9090' sending message
2012-12-01 21:00:19,327 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,61} [d=0.0165,l=1.59999e-05,b=1e+07]
2012-12-01 21:00:19,327 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #41
2012-12-01 21:00:19,327 DEBUG [comet.rae.tcp_connection] '[3105::8]:10001' sending message
2012-12-01 21:00:19,329 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,62} [d=0.016,l=1.49999e-05,b=1e+07]
2012-12-01 21:00:19,333 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #63
2012-12-01 21:00:19,334 DEBUG [comet.rae.cme_connection] '[2001:67c:24cc:31c1::ae:1]:9090' sending message
2012-12-01 21:00:19,577 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,62} [d=0.0165,l=1.59999e-05,b=1e+07]
2012-12-01 21:00:19,577 DEBUG [comet.rae.rae_logic] best paths: length=4 {64,63,61,62} [d=0.0215,l=2.09998e-05,b=1e+07]
2012-12-01 21:00:19,577 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #61
2012-12-01 21:00:19,577 DEBUG [comet.rae.rae_logic] updating prefix 2001:67c:24cc:31f1::/64 in class #1
2012-12-01 21:00:19,577 DEBUG [comet.rae.tcp_connection] '[2001:67c:24cc:31c1::ae:1]:10001' received message
2012-12-01 21:00:19,577 DEBUG [comet.rae.tcp_connection] '[2001:67c:24cc:31e1::ae:3]:10001' received message
2012-12-01 21:00:19,599 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,62} [d=0.0165,l=1.59999e-05,b=1e+07]
2012-12-01 21:00:19,599 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #51
2012-12-01 21:00:19,599 DEBUG [comet.rae.tcp_connection] '[2a02:9008:0:1911:0:50:56a3:51]:56413' sending message
2012-12-01 21:00:19,600 DEBUG [comet.rae.rae_logic] best paths: length=4 {64,63,61,62} [d=0.022,l=2.19998e-05,b=1e+07]
2012-12-01 21:00:19,620 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #63
2012-12-01 21:00:19,621 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,62} [d=0.016,l=1.49999e-05,b=1e+07]
2012-12-01 21:00:19,621 DEBUG [comet.rae.rae_logic] best paths: length=4 {64,63,61,62} [d=0.022,l=2.19998e-05,b=1e+07]
2012-12-01 21:00:19,628 DEBUG [comet.rae.rae_logic] best paths: length=4 {51,62,63,64} [d=0.1165,l=2.39998e-05,b=1e+07]
2012-12-01 21:00:19,628 DEBUG [comet.rae.rae_logic] best paths: length=5 {51,41,61,63,64} [d=0.2215,l=0.00103097,b=1e+07]
2012-12-01 21:00:19,628 DEBUG [comet.rae.rae_logic] best paths: length=5 {51,62,61,63,64} [d=0.117,l=2.69997e-05,b=1e+07]
2012-12-01 21:00:19,643 DEBUG [comet.rae.cme_connection] '[2001:67c:24cc:31d1::ae:2]:9090' received message
2012-12-01 21:00:19,643 DEBUG [comet.rae.cme_connection] '[2001:67c:24cc:31d1::ae:2]:9090' sending message
2012-12-01 21:00:19,644 DEBUG [comet.rae.cme_connection] '[2001:67c:24cc:31d1::ae:2]:9090' received header, message size=29
2012-12-01 21:00:19,644 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,62} [d=0.0165,l=1.59999e-05,b=1e+07]
2012-12-01 21:00:19,644 DEBUG [comet.rae.rae_logic] best paths: length=4 {64,63,61,62} [d=0.0215,l=2.09998e-05,b=1e+07]
2012-12-01 21:00:19,644 DEBUG [comet.rae.rae_logic] best paths: length=6 {64,63,61,41,51,62} [d=0.1475,l=5.39988e-05,b=1e+07]
2012-12-01 21:00:19,644 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #61
2012-12-01 21:00:19,644 DEBUG [comet.rae.rae_logic] updating prefix 2001:67c:24cc:31f1::/64 in class #1
2012-12-01 21:00:19,644 DEBUG [comet.rae.tcp_connection] '[2a02:9008:0:1911:0:50:56a3:51]:56413' received message
2012-12-01 21:00:19,675 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #51
2012-12-01 21:00:19,677 DEBUG [comet.rae.rae_logic] best paths: length=3 {61,63,64} [d=0.011,l=1.2e-05,b=1e+07]
2012-12-01 21:00:19,734 DEBUG [comet.rae.rae_logic] updating prefix 2001:67c:24cc:31f1::/64 in class #1
2012-12-01 21:00:19,735 DEBUG [comet.rae.rae_logic] best paths: length=2 {64,63} [d=0.006,l=6.99999e-06,b=1e+07]
2012-12-01 21:00:19,735 DEBUG [comet.rae.tcp_connection] '[2001:67c:24cc:31c1::ae:1]:54489' sending message
2012-12-01 21:00:19,736 DEBUG [comet.rae.cme_connection] '[2001:67c:24cc:31c1::ae:1]:9090' sending message
2012-12-01 21:00:19,736 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,61} [d=0.0165,l=1.59999e-05,b=1e+07]
2012-12-01 21:00:19,736 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #62
2012-12-01 21:00:19,736 DEBUG [comet.rae.rae_logic] updating prefix 2001:67c:24cc:31f1::/64 in class #1
2012-12-01 21:00:19,736 DEBUG [comet.rae.tcp_connection] '[2001:67c:24cc:31d1::ae:2]:59994' received message
2012-12-01 21:00:19,737 DEBUG [comet.rae.rae_logic] best paths: length=4 {64,63,62,61} [d=0.0215,l=2.09998e-05,b=1e+07]
2012-12-01 21:00:19,746 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #62
2012-12-01 21:00:19,746 DEBUG [comet.rae.tcp_connection] '[2001:67c:24cc:31d1::ae:2]:38509' sending message
2012-12-01 21:00:19,747 DEBUG [comet.rae.rae_logic] best paths: length=2 {64,63} [d=0.006,l=6.99999e-06,b=1e+07]
2012-12-01 21:00:19,758 DEBUG [comet.rae.cme_connection] '[2001:67c:24cc:31e1::ae:3]:9090' sending message
2012-12-01 21:00:19,784 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #63
2012-12-01 21:00:19,785 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,62} [d=0.016,l=1.49999e-05,b=1e+07]
2012-12-01 21:00:19,785 DEBUG [comet.rae.rae_logic] best paths: length=4 {64,63,61,62} [d=0.022,l=2.19998e-05,b=1e+07]
2012-12-01 21:00:19,785 DEBUG [comet.rae.rae_logic] best paths: length=6 {64,63,61,41,51,62} [d=0.1475,l=5.39988e-05,b=1e+07]
2012-12-01 21:00:19,788 DEBUG [comet.rae.rae_logic] best paths: length=2 {64,63} [d=0.003,l=3e-06,b=2e+07]
2012-12-01 21:00:19,876 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #41
2012-12-01 21:00:19,877 DEBUG [comet.rae.rae_logic] best paths: length=3 {62,63,64} [d=0.005,l=4.99999e-06,b=1.6e+07]
2012-12-01 21:00:19,877 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,61} [d=0.0165,l=1.59999e-05,b=1e+07]
2012-12-01 21:00:19,877 DEBUG [comet.rae.rae_logic] best paths: length=4 {62,61,63,64} [d=0.006,l=5.99998e-06,b=2e+07]
2012-12-01 21:00:19,877 DEBUG [comet.rae.rae_logic] best paths: length=4 {64,63,62,61} [d=0.022,l=2.19998e-05,b=1e+07]
2012-12-01 21:00:19,877 DEBUG [comet.rae.rae_logic] best paths: length=6 {62,51,41,61,63,64} [d=0.118,l=0.000109999,b=1.6e+07]
2012-12-01 21:00:19,899 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #63
2012-12-01 21:00:20,271 DEBUG [comet.rae.rae_logic] updating prefix 2001:67c:24cc:31f1::/64 in class #1
2012-12-01 21:00:20,309 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,61} [d=0.0165,l=1.59999e-05,b=1e+07]
2012-12-01 21:00:20,309 DEBUG [comet.rae.rae_logic] best paths: length=4 {64,63,62,61} [d=0.022,l=2.19998e-05,b=1e+07]
2012-12-01 21:00:20,309 DEBUG [comet.rae.rae_logic] best paths: length=6 {64,63,62,51,41,61} [d=0.232,l=0.00103996,b=1e+07]
2012-12-01 21:00:20,309 DEBUG [comet.rae.rae_logic] propagating prefix 2001:67c:24cc:31f1::/64 in class #1 for source #41
2012-12-01 21:00:20,339 DEBUG [comet.rae.rae_logic] best paths: length=3 {64,63,61} [d=0.016,l=1.49999e-05,b=1e+07]
2012-12-01 21:00:20,339 DEBUG [comet.rae.rae_logic] best paths: length=4 {64,63,62,61} [d=0.022,l=2.19998e-05,b=1e+07]
2012-12-01 21:00:20,339 DEBUG [comet.rae.rae_logic] best paths: length=6 {64,63,62,51,41,61} [d=0.2325,l=0.00104096,b=1e+07]
```

Figure 101 Exemplary log file collected from RAEs after prefix advertisement

Figure 102 presents the values of RCT measured for prefix advertisement and withdrawal, while Table 22 summarises basic statistics of RCT.

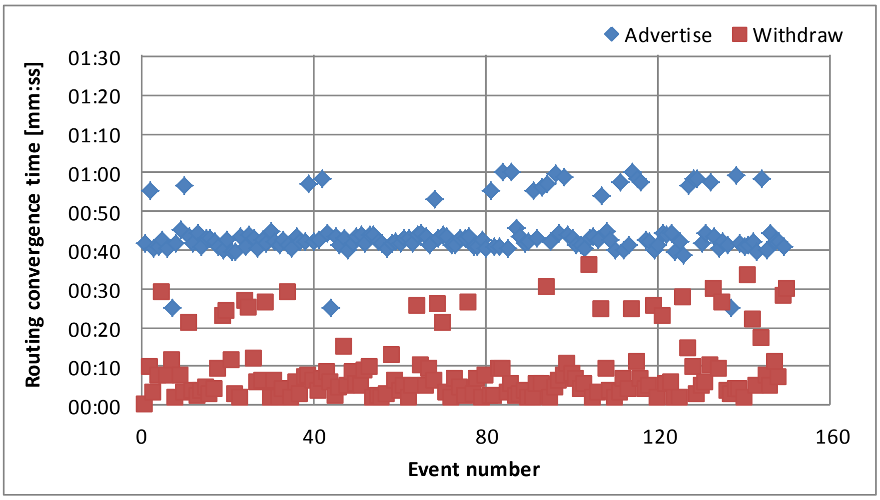


Figure 102 Values of RCT for prefix advertisement/withdrawal in stub domain

Statistics of RCT	Prefix advertisement [mm:ss.000]	Prefix withdrawal [mm:ss.000]
mean	00:44.409	00:08.977
std	00:06.472	00:08.605
95% percentile	00:58.672	00:27.927

Table 22: Statistics of RCT for prefix advertisement/withdrawal in stub domain

We can observe that:

- The RCT after prefix advertisement took about 44s with the 95 percentile limit at 58s. This relatively large value comes from advertisement of a number of alternative paths. The alternative paths are calculated and propagated in sequence. Once alternative path is discovered, the RAE starts to propagate this information. Moreover, RAEs are protected by Minimum Route Advertisement Interval (MRAI) timer which defines minimum interval between consecutive advertisements related to the same prefix. The MRAI timer prevents route oscillations and saves processing power but may increase the convergence time [12]. Note that the prefix advertisement process is less time critical than prefix withdrawal.
- The RCT after prefix withdrawal took about 8 s with the 95 percentile at 27 s. These values are significantly lower comparing to prefix advertisement. They are mainly determined by the message processing and relatively large transfer delays between sides of the federated testbed.

In PER-RAE-003 test, we introduced sequence of link failures and repairs on the inter-domain link between domain 61 and 63. Once the link failure/repair occurs, the RAEs have to detect this event and then recalculate routing paths. For each stressing event, we measure the RCT and check correctness of RAE routing table. The presented procedure was repeated about 150 times. Again, we assume 5 min interval between consecutive prefix advertisement and withdrawal events to reach stable routing in the federated testbed.

Figure 103 presents the values of RCT measured for link failure and repair, while Table 23 summarises basic statistics of RCT.

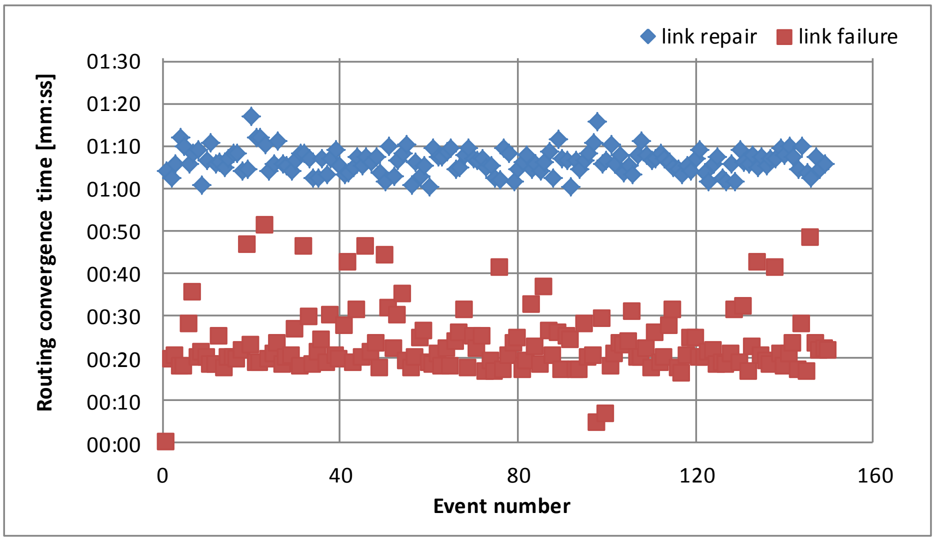


Figure 103 Values of RCT for link failure/repair occurring in the inter-domain link

Statistics of RCT	link failure [mm:ss.ooo]	link repair [mm:ss.ooo]
mean	00:24.021	01:06.545
std	00:08.110	00:03.929
95% percentile	00:45.163	01:11.065

Table 23: Statistics of RCT for link failure and repair

We can observe that:

- The RCT after link repair took 1 min and 6 sec with the 95 percentile limit at 1 min and 11s. Similar as in the case of prefix advertisements, this relatively large value comes from advertisement of a number of alternative paths. The alternative paths are calculated and propagated in sequence. Once alternative path is discovered, the RAE starts to propagate this information. In case of link repair between core domains, a number of new alternative paths have to calculated, so the convergence process takes slightly longer that in case of prefix advertisement in stub domain. Moreover, RAEs are protected by Minimum Route Advertisement Interval (MRAI) timer which defines minimum interval between consecutive advertisements related to the same prefix. The MRAI timer prevents route oscillations and saves RAE processing power but may increase the convergence time [12].
- The RCT after link failure took 24s with the 95 percentile at 45s. In this case, the RAE have to withdraw all affected paths and switch routing to alternative paths.

The tests performed in the federated testbed confirmed that RAE correctly calculates routing paths. Moreover, the RCT measured for basic stressing events, i.e. prefix advertisement/withdrawal, link failure/repair are at similar level as BGP-4 convergence times reported in [12]. However, the RCT strongly depends on the number of domains, network topology, processing power of routers, transmission delays. Taking into account limited topology of federated testbed, which covers only 6 domains, the obtained results should be treated rather as exemplary, illustrating RAE behaviour in distributed network environment. Therefore, the RAE performance analysis should be complemented by simulation studies performed in large scale scenarios.

4.2.5 CAFE Tests

This section presents CAFE performance tests. They focus on evaluation of maximum packet forwarding capabilities offered by edge and core CAFEs as well as the performance of the CME-CAFE interface and the edge CAFE configuration latency.

4.2.5.1 Tests for Edge CAFE

4.2.5.1.1 PER-CAFE-001

The objective of this test is to evaluate the maximum packet forwarding throughput of edge CAFE. The throughput metric is defined as the maximum data rate of packet stream forwarded without packet losses. We assume that Device under Test (DUT) forwards packets without losses, when packet loss ratio is smaller than 10^{-5}. In order to assess obtained results, we compare CAFE performance with the reference IPv6 software router running on the same PC machine. Moreover, we monitor the CPU load to get knowledge about host status.

Let us recall that packet forwarding inthe edge CAFE incorporates two functions:

- Interception and encapsulation of packets within COMET header. This action is performed by *cafe_intercept* module, see [8] for details,
- Forwarding of frames based on COMET header. This action is performed by *cafe_forward* module, see [8] for details.

Taking into account that federated testbed is deployed on virtual machines that have limited processing power and memory, the PER-CAFE-001 test was performed in the standalone testbed. It consists of Device under Test (DUT) and Automatic Test Equipment (ATE) interconnected by two 1Gbps links as depicted on Figure 104.

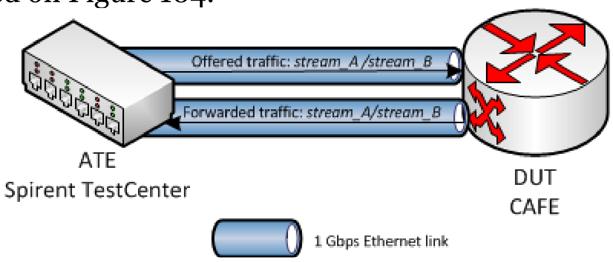


Figure 104 Topology assumed for PER-CAFE-001 test

The edge CAFE was installed on ordinary PC machine with Intel(R) Core(TM) 2 Duo CPU E8500 @ 3.16GHz processor and 16 GB of RAM memory equipped with Intellinet gigabit PCI express network cards (model 522533).The testing tool was the Spirent TestCenter equipped with hardware traffic generator/analyser card CM-1G-D4.

The packet forwarding throughput is measured for IPv6 packets carried over Ethernet. Packets are intercepted on the basis of: source IP address, destination IP address, transport protocol, source port and destination port. The intercepted packets are encapsulated within COMET header containing five forwarding keys. The total length of COMET header is equal to 7 octets. Our tests are performed for the following size of packets: 128B, 256B, 512B, 768B, 1016B, 1024B, 1032B, 1256B and 1512B (the frame size includes Ethernet, IPv6, UDP headers and payload, it does not contain Ethernet CRC checksum).Each test is performed for at least 10^6packets forwarded by the edge CAFE. The forwarding throughput for edge CAFE is estimated with 1Mbps accuracy.

The testing procedure follows methodology proposed in RFC 2544 [13]and constitutes of following steps:

- Prerequisites:
    - Configure Spirent TestCenter with two streams of IPv6 traffic (i.e. *stream_A* and *stream_B*).

- Configure edge CAFE to intercept packets from stream_A, encapsulate them with COMET header and forward to output interface.
- Configure IPv6 router to forward packets from *stream_B*to output interface

- Testing (performed for each frame size):
    - Increase the offered load of *stream_A* traffic (no traffic from *stream_B*) until observing packet losses in the received packet stream that are equal or greater than 10^{-5}. Record maximum rate of offered traffic, corresponding rate of forwarded traffic and CPU load on machine.
    - Increase the offered load of *stream_B* traffic (no traffic from *stream_A*) until observing packet losses in the received packet stream that are equal or greater than 10^{-5}. Record maximum rate of offered traffic and CPU load on machine.

The obtained results are presented in Figure 105.

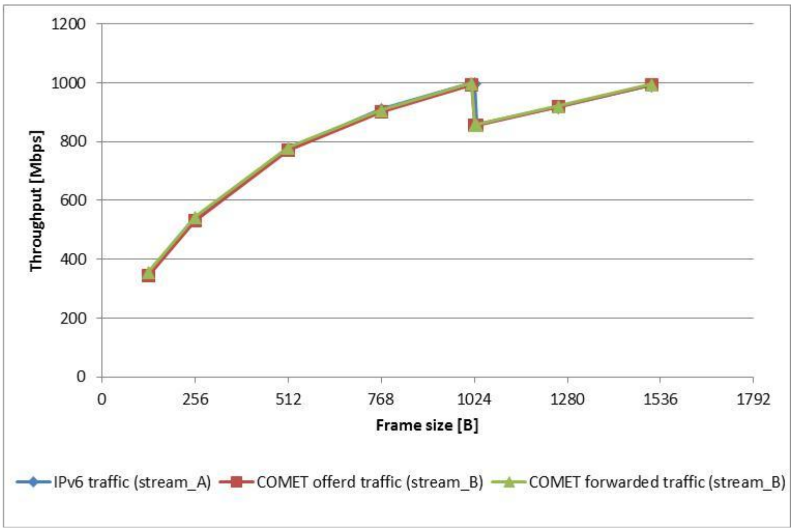


Figure 105 PER-CAFE-001 – forwarding throughput

The throughput of edge CAFE is similar to the throughput of IPv6 software router, see details in Table 24. The small differences come from slightly more complicated operations performed by edge CAFE, which additionally covers packet interception and encapsulation. Moreover, due to COMET encapsulation with additional header (7 bytes), the rate of outgoing stream is slightly higher than the rate of offered stream. During all tests the CPU load has not exceeded 3%,so we conclude that CPU power is not a bottleneck in this case.

We can observe that throughput of both *streams* increases with the frame size. However, there is unexpected performance decrease effect for frame size slightly larger than 1024B. This effect comes from constraints of PC hardware, which is typically optimized for 1024B frames. Nevertheless, for all range of frame sizes throughput of COMET forwarding is at the same level as throughput of IPv6 forwarding. Moreover, throughput of *stream_B* traffic is between offered traffic rate for *stream_A*, and forwarded traffic rate for *stream_A*.

Frame size[B] of IPv6 traffic and COMET offered traffic	Frame size[B] of COMET forwarded traffic	IPv6 traffic [Mbps]	COMET offered traffic [Mbps]	COMET forwarded traffic [Mbps]
128	135	350	343	359
256	263	537	532	546
512	519	774	769	779
768	775	912	900	908

Frame size[B] of IPv6 traffic and COMET offered traffic	Frame size[B] of COMET forwarded traffic	IPv6 traffic [Mbps]	COMET offered traffic [Mbps]	COMET forwarded traffic [Mbps]
1016	1023	1000	993	1000
1024	1031	1000	854	860
1032	1039	858	854	860
1256	1263	918	918	923
1512	1519	992	991	996

Table 24: PER-CAFE-001 – forwarding throughput

4.2.5.1.2 PER-CAFE-002

The objective of this test is to evaluate the ability of edge CAFE to intercept and handle large number of running flows. We performed two test cases: test case #1 evaluates of edge CAFE throughput as a function of number of running flows. The test case #2 evaluates the impact of packet size on edge CAFE throughput under large number of running flows.

The topology assumed for both test cases is depicted in Figure 106. We use the same equipment as in previous test described in section 4.2.5.1.1.

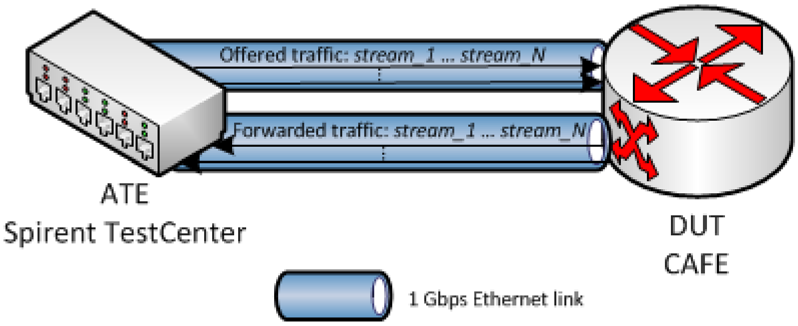


Figure 106 Topology assumed for PER-CAFE-002test

In the first test case, we measure the edge CAFE throughput as a function of a number of running flows. All measurements were performed for frame size equal to 1016B, i.e. frame size for which edge CAFE gets the best performance (see 4.2.5.1.1 for details)

The testing procedure covers the following steps:

- Configure Spirent TestCenter to generate N streams of IPv6 traffic. Each stream, identified by unique source IPv6 address, is configured the send traffic with the same rate.
- Configure edge CAFE to: intercept N streams, encapsulate them within the COMET header and forward them to output interface.
- Measure throughput and CPU load on edge CAFE.

The Table 25 presents the obtained results. In all measurements, the CPU load was negligible, i.e. less than 4%.

Number of flows	COMET offered traffic [Mbps]	COMET forwarded traffic [Mbps]
1	991	1000
10	991	1000
100	991	1000
1000	991	1000
10000	991	1000

Table 25: PER-CAFE-002 – case #1 test results

We conclude that the edge CAFE is able to intercept and forward 1 Gbps traffic with the number of running flows in range 1 – 10000.

In the second test case, we assess whether edge CAFE is able to forward traffic with maximum throughput for 10000 simultaneously running flows.

The testing procedure covers the following steps:

- Configure Spirent TestCenter to generate 10000 streams of IPv6 traffic. Each stream, identified by unique source IPv6 address is configured the send traffic with the same rate.
- Configure edge CAFE to: intercept 10000 streams, encapsulate them with COMET header and forward them to output interface.
- Measure throughput and CPU load on edge CAFE.

Table 26 presents obtained results.

Frame size[B] of IPv6 traffic (COMET offered traffic)	Frame size[B] of COMET forwarded traffic	Number of flows	COMET offered traffic [Mbps]	COMET forwarded traffic [Mbps]	Mean CPU load [%]
128	135	10000	343	359	18.1
512	519	10000	769	779	6.7
1016	1023	10000	993	1000	3.7
1512	1519	10000	991	996	2.8

Table 26: PER-CAFE-002 – case #2 test results

We conclude that edge CAFE is able to forward packets from 10000 simultaneously running flows. The results say the CPU load of edge CAFE is inversely proportional to frame size. This effect is caused by the per frame lookup in the COMET interception table.

4.2.5.2 Tests for core CAFE

4.2.5.2.1 PER-CAFE-003

The objective of this test is to evaluate forwarding throughput of core CAFE. Similar to test PER-CAFE-001, we measure throughput metric, which defines the maximum data rate of packet stream forwarded by core CAFE without packet losses. Again, in order to assess obtained results, we compare core CAFE performance with the reference IPv6 software router running on the same PC machine. The packet forwarding on core CAFE is performed by cafe_forward module based on COMET header, see [8] for details.

The tests are performed in a standalone testbed depicted on Figure 107. It consists of Spirent TestCenter with CM-1G-D4 card (Automatic Test Equipment - ATE) and Linux machine (Device under Test - DUT) interconnected by two 1Gbps links as. For the DUT we use ordinary PC machine with Intel(R) Core(TM) 2 Duo CPU E8500 @ 3.16GHz and 16 GB of RAM memory equipped with Intellinet gigabit PCI express network cards (model 522533).

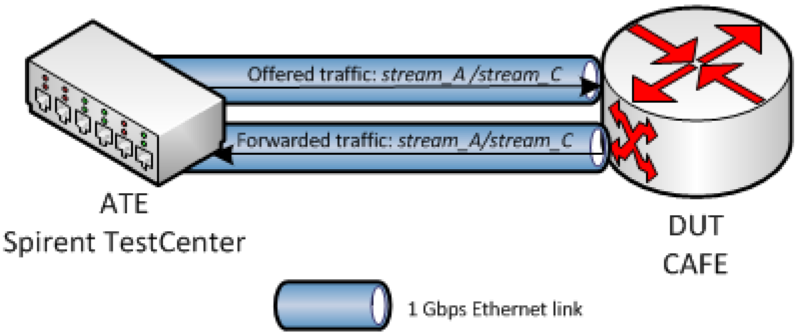


Figure 107PER-CAFE-003- topology

The packet forwarding throughput is measured for COMET frames carried over Ethernet. The tests are performed for frames of the following sizes: 128B, 256B, 512B, 768B, 1024B, 1032B, 1256B and 1512B. The frame size includes Ethernet, COMET header, IPv6, UDP headers and payload. It does not contain Ethernet CRC checksum.

The performance of forwarding throughput of COMET packets on core CAFE is compared to performance of forwarding throughput of standard IPv6 packets (tests are performed on the same physical machine). Each test is performed for at least 10^6 frames forwarded by the core CAFE. The forwarding throughput for core CAFE is estimated with 1Mbps accuracy. Moreover for each test the CPU load is measured.

The testing procedure follows methodology proposed in RFC 2544 [13]and constitutes of following steps:

- Prerequisites:
  - Configure Spirent TestCenter with one stream of IPv6 traffic (i.e. *stream_A*).
  - Configure Spirent TestCenter with one stream of COMET traffic (i.e. *stream_C*)
  - Configure routing in core CAFE to forward packets from *stream_C* to output interface
  - Configure IPv6 routing table to forward packets from stream A to output interface.
- Testing (performed for each frame size):
  - Increase the offered load of *stream A* traffic (no traffic from *stream_C*) until observing packet losses in the received packet stream that are equal or greater than 10^{-5}. Record maximum rate of offered traffic and CPU load on machine.
  - Increase the offered load of *stream_C* traffic (no traffic from *stream A*) until observing packet losses in the received packet stream that are equal or greater than 10^{-5}. Record maximum rate of offered traffic and CPU load on machine.
  -

The obtained results are presented in Figure 108.

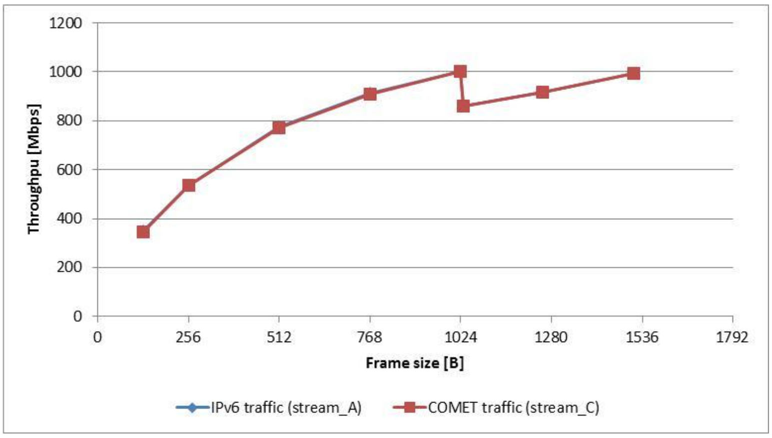


Figure 108 PER-CAFE-003 – forwarding throughput

The throughput of core CAFE is the same as throughput of IPv6 router, see details in Table 27. During all tests the CPU load didn't exceeded 3%, so we conclude that CPU power is not a bottleneck in this case.

Frame size[B]	IPv6 traffic [Mbps]	COMET offered traffic [Mbps]
128	350	347
256	537	535
512	774	771
768	912	910
1024	1000	1000
1032	858	858
1256	918	918
1512	992	992

Table 27: PER-CAFE-003 – forwarding throughput

We can observe that throughput of *stream_C* and throughput of *stream_A* increase with the frame size increases. However, we can observe unexpected performance decrease effect for frame size about 1024. This effect is similar to the results of edge CAFE (similarly to point 4.2.5.1.1). Moreover, throughput of core CAFE is lightly higher than throughput of edge CAFE, because core CAFE does not perform encapsulation with COMET header.

4.2.5.2.2 PER-CAFE-004

The objective of this test is to evaluate the performance of core CAFE for increasing number of running flows. We performed two test cases: case #1 - evaluation of core CAFE forwarding performance in function of a number of running flows, and case #2 evaluation of core CAFE forwarding in function of frame size. The test topology for both cases is depicted in Figure 106. The test equipment for cases is the same as described in 4.2.5.1.1.

In the first test case we assess whether the core CAFE is able to forward packets with maximum throughput independently on number of running flows. All measurements in this test case were performed for frame size equal to 1024B, i.e. frame size for which core CAFE gets the best performance (see Table 27for details)

The testing procedure covers the following steps:

- Configure Spirent TestCenter with N streams of COMET traffic, each stream (identified by unique list of COMET keys) sends COMET packets at the same rate.
- Configure core CAFE to forward all frames with COMET header to output interface.
- Measure the forwarded traffic rate and record the CPU load of edge CAFE.

Notice that, as each forwarding key in COMET header is one octet long than the maximal number of entries in COMET forwarding table is equal to 256 (2^8).

The Table 28 presents obtained results. In all measurements the CPU load was negligible, i.e. it is less than3%.

Number of flows	COMET offered traffic [Mbps]	COMET forwarded traffic [Mbps]
1	1000	1000
10	1000	1000
100	1000	1000
1000	1000	1000
10000	1000	1000

Table 28: PER-CAFE-004 – case #1 test results

We conclude that edge CAFE is able to forward 1 Gbps of traffic up to 10 000 running flows.

In the second test case, we assess the impact of frame size on edge CAFE handling 10000 simultaneous flows.

The testing procedure covers the following steps:

- Configure Spirent TestCenter with 10000 streams of COMET traffic. Each stream, identified by unique list of COMET keys, is configured the send traffic at the same rate.
- Configure core CAFE to forward all frames with COMET header to output interface.
- Measure the forwarded traffic rate and record the CPU load of edge CAFE.

The Table 29 presents obtained results.

Frame size[B]	COMET offered traffic [Mbps]	COMET forwarded traffic [Mbps]	Mean CPU load [%]
128	347	347	6
512	771	771	3.8
1024	1000	1000	3
1512	992	992	2.3

Table 29: PER-CAFE-004 – case #2 test results

We conclude that edge CAFE is able to forward packets from 10000 simultaneously running flows of considered sizes. (for each frame size, COMET forwarded traffic rates for 1 flow and for 10000 flows are equal, see Table 29 and Table 26). Moreover, the mean CPU load is of edge CAFE inversely proportional to frame size. This effect is caused by the per frame lookup procedure in COMET forwarding table.

4.2.5.3 Tests on CME-CAFE interface

4.2.5.3.1 PER-CAFE-005

The objective of this test is to evaluate time required for configuration of interception filter on edge CAFE through the CME-CAFE interface (see for [8] details). This metric has impact on the content t retrieval time; therefore measured values will be used in scalability studies. The test is performed in two limit cases, which differ in arrival process of configuration requests. The case #1 assumes that configuration requests arrive sporadically, while the case #2 assumes greedy arrival of configuration requests. The test topology for both cases is depicted in Figure 109.

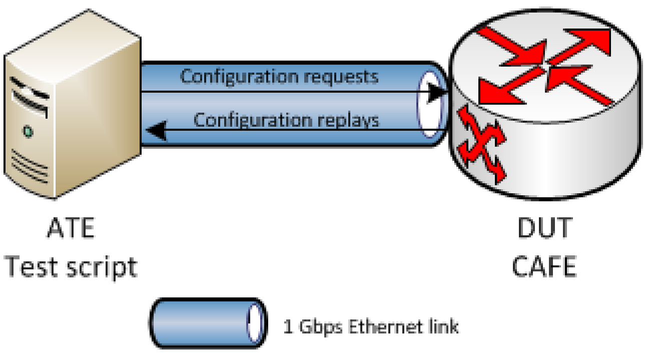


Figure 109 PER-CAFE-005, test topology

The ATE (Automated Test Equipment) is connected with CAFE (acting as DUT) with 1 Gbps direct link. The ATE sends configuration requests to *cafe_agent* (see [8] for details) running on CAFE machine. Upon receiving requests, *cafe_agent* configures interception filter in *cafe_intercept* module and responses to ATE with information about configuration status: success or failure. The connection between ATE and CAFE exploits TCP protocol (similarly to connection between CME and CAFE). The CAFE configuration time is defined as the time between sending the configuration request by ATE and receiving the reply. It covers: request handling time in an operating system at ATE and DUT machines, transmission time of request and reply messages, propagation delays connection between ATE and CAFE exploits 1m Ethernet cable, and finally, configuration time of filtering at edge CAFE.

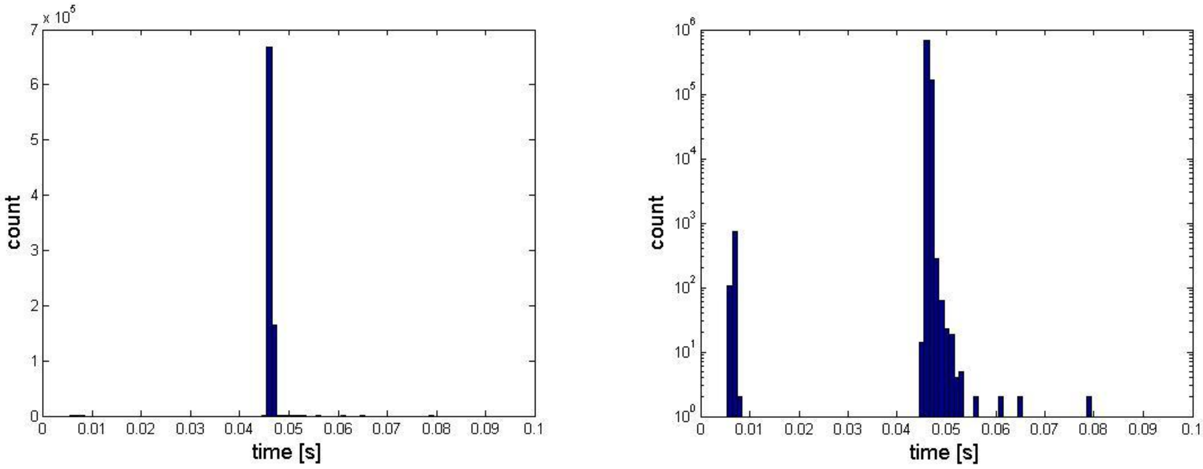
In the test case #1, configuration requests are sent sporadically with large inter-arrival times, i.e. next configuration request is sent after ATE receives configuration reply corresponding to the previous configuration request and an idle period expires. In the test case #2, requests are sent in parallel, i.e. next configuration request is sent just after sending the previous one. So,the ATE doesn't wait to receive response for previous configuration request before sending the next configuration request. The first test case is used to measure the CAFE configuration time for sporadic requests, whereas the second test case is used to measure the maximum rate that CAFE can successfully handle configuration requests. For both cases, special tools have been developed to measure the configuration latency.

In order to mitigate measurement impairments introduced by ATE, we exploited relatively modern PC for ATE (to execute test script), and relatively slow PC for CAFE. The test script is run on PC machine with Intel(R) Core(TM)4 Duo CPU E8500 CPU @ 3.16GHz and 16 Gb of RAM memory, whereas the CAFE is run on PC machine with Intel(R) Pentium (R) 2CPU @ 3.00E GHz and 512Mb of RAM memory. Both machines are equipped with Intellinet gigabit PCI express network cards (model 522533).

In the case#1, the ATE sent 85000 sporadic configuration requests and measured the edge CAFE response time. The Table 30 presents configuration time statistics, whereas Figure 110 presentshistogram of configuration time in linear and logarithmic scale. The histograms are calculated with 1ms buckets.

Metric	Latency [s]
Minimal configuration time	6,20 * 10^{-3}
Maximal configuration time	79,21 * 10^{-3}
Mean configuration time	46,00 * 10^{-3}
Standard deviation of configuration time	1,30 * 10^{-3}

Table 30: PER-CAFE-005, case #1: sporadic requests



(a) linear scale (b) logarithmic scale

Figure 110 Histogram of CAFE configuration time: (a) linear scale (b) logarithminc scale.

In case of sporadic requests the mean configuration time of edge CAFE equals 46ms with relatively small standard deviation equal to 1.3ms. The observed variation is caused by variable waiting time for CPU interrupt both in ATE and DUT. The minimum configuration time equals 6.2ms and corresponds to the situation where these processes at ATE and DUT get the CPUs interrupt immediately upon request. On the other hand, the maximum configuration time equals 79.21ms, and corresponds to situation when at least one of these processes have to wait for CPU to finish running task.

In test case#2, ATE sent 10 000 consecutive configuration requests and measured CAFE response time between sending first configuration request and receiving last configuration reply. The testing tool prepares 10 000 configuration requests in advance and then sends them to DUT as quick as possible using a TCP connection.

The obtained results are presented in Table 31.

Result	**Value**
Tests duration [s]	78.3973
Number of requests:	10000
Successful configurations	10000
Failed configurations	0
CAFE performance [requests/s]	127.5552
CAFE mean response time [s]	0.0078

Table 31: PER-CAFE-005, case #2: consequtive requests

All configuration requests were successfully configured in about 78 seconds, with average rate of about 127 requests per second. The mean response time of CAFE was about 8ms, what is similar value to the minimum value of CAFE response time measured for sporadic requests. Notice that in test case #1, the mean configuration time was about 46ms. This much longer time is caused by the fact that in test case #1 measurement processes (at ATE and DUT) have to wait for CPUs interrupts for each configuration request, whereas in test case #2 these waiting times corresponded to whole batches of requests (a number of requests). In test case #2, measurement processes after receiving interrupt form CPU processed a batch of requests. As a consequence, the results of the test case #1 correspondto the worst case situation, while the results of the test case #2 correspond to best case situation in terms of CAFE response times.

5 Requirements and Performance Metrics Validation

This sections aims at verifying in that main qualitative requirements of D2.1 have been addressed (see Section 4.3.1), validated and verified and in that the selected performance metrics addressed as quantitative requirements in D6.1 (see Section 4.3.2) have been satisfied.

5.1 Qualitative Requirement Validation and Verification

ID	Category	System requirement	Test IDs	Validated (Comments)
1	**Global**	Content as a primitive	FUN-UC1-003 FUN-UC2-003 FUN-UC1-014 to FUN-UC1-015 FUN-UC1-019 FUN-UC4-002	YES
2	**Global**	Global content naming and addressing	PER-PUB-001 Partially Applicable in WP5	YES
3	**Global**	Open for future evolution of the Internet	Not Applicable in WP6 (architectural requirement)	N/A
4	**Global**	Scalable to be deployed in the largest ISPs	Not Applicable (WP5)	N/A
5	**Global**	Involvement of all Internet users as Content Creators	FUN-UC1-002	YES
6	**Global**	Graceful switching of the content delivery path without impact on the application-layer	Not covered by the selected Use Cases	N/A
7	**Content Consumer**	Access independent from content location	FUN-UC1-011 FUN-UC1-014 FUN-UC1-015	YES
8	**Content Consumer**	Content ID independent from way distribution and nature of content	FUN-UC1-010 FUN-UC1-012 FUN-UC1-013 FUN-UC1-014 FUN-UC1-015 FUN-UC1-019 FUN-UC4-002 FUN-UC4-004	YES
9	**Content Consumer**	User unawareness	FUN-UC1-003 FUN-UC1-008 FUN-UC1-013 FUN-UC1-019 FUN-UC4-003	YES
10	**Content Consumer**	Content Client able to declare his capabilities	FUN-UC1-003 to FUN-UC1-008 FUN-UC1-014 to FUN-UC1-018 FUN-UC4-001	YES

ID	Category	System requirement	Test IDs	Validated (Comments)
11	**Content Consumer**	Content Client will obtain all necessary parameters to invoke the application level requests	FUN-UC1-008 FUN-UC1-013 FUN-UC4-003 FUN-UC1-019 FUN-UC4-003	YES
12	**Content Provider**	Interface to update the content properties	FUN-UC1-010 FUN-UC1-012	YES
13	**Content Provider**	Capability of establishing policies to enforce the way to deliver contents	FUN-UC1-002 FUN-UC1-010 FUN-UC1-012 FUN-UC14 to FUN-UC-16 FUN-UC4-002	YES
14	**CMP**	Global content resolution architecture	FUN-UC1-004	YES
15	**CMP**	Integrated traffic and resource management solution to increase network efficiency and content delivery	FUN-UC1-001 FUN-UC1-004 FUN-UC1-014 to FUN-UC1-18 FUN-UC4-001 FUN-UC4-004	YES
16	**CMP**	Information gathering system	FUN-UC1-005 FUN-UC1-009 FUN-UC1-014 FUN-UC1-018 FUN-UC1-019 PER-RAE-001 PER-RAE-002 PER-RAE-003	YES
17	**CMP**	Efficient protocol interfaces	FUN-UC1-002 FUN-UC1-003 FUN-UC1-008 FUN-UC1-014 to FUN-UC1-018 FUN-UC4-002 FUN-UC4-003	YES
18	**CMP**	Capability of dynamically modify servers location information	FUN-UC1-010 FUN-UC1-012 FUN-UC1-014 to FUN-UC1-018 FUN-UC4-002	YES
19	**CMP**	Possibility of registering different ways of distribution	FUN-UC1-010 FUN-UC1-012 FUN-UC1-014 to FUN-UC1-018 FUN-UC4-002	YES

ID	Category	System requirement	Test IDs	Validated (Comments)
20	CMP	Network conditions and routing information awareness	Only for long term information FUN-UC1-001 FUN-UC4-001 PER-RAE-001 PER-RAE-002 PER-RAE-003	YES
21	CMP	Interaction between the Content Mediation Servers and the Content Aware Forwarders to enforce content delivery	FUN-UC1-005 FUN-UC1-007 FUN-UC1-014 to FUN-UC1-018 FUN-UC4-003 PER-CAFE-005	YES
22	CMP	CMP able to request the enforcement of QoS and multicast in the network	FUN-UC1-014 to FUN-UC1-018 FUN-UC1-019	YES
23	CFP	Content forwarding architecture able to reach IP-based forwarding speeds	PER-CAFE-001 to PER-CAFE-004	YES
24	CFP	Elements in CFP able to support QoS-aware content delivery	FUN-UC1-001 FUN-UC1-014 to FUN-UC1-018 FUN-UC1-001	YES
25	CFP	Elements in CFP able to support point-to-multipoint content delivery	FUN-UC1-019 FUN-UC1-020	YES
26	CFP	Content may be cached to optimize network resource usage	Not in decoupled Approach	YES
27	CFP	Interaction between the CFP and the CMP to provide information on network conditions and, optionally, routing information	Only long term info FUN-UC1-001 FUN-UC1-001	YES

Table 32: Mapping of Qualitative Requirements to test cases

5.2 Quantitative Requirements Validation and Verification

Metric name	Reference value	TestID	Validated (comments)
Content Retrieval Latency (CRL)	95 percentile of CRL: < 213 ms, for single domain case < 3 s, as users' tolerable limit	PER-CME-01 Only small scale, large scale will be covered in WP5	CME CRT time was around 110ms under sensible request load.
Content Resolution Time (CRT)	95 percentile of CRT: < 130 ms, for single domain case < 2.5s, as users' tolerable limit	PER-CRE-02 Only small scale, large scale will be covered in WP5	The resolution time was <10ms.

Metric name	Reference value	TestID	Validated (comments)
Content retrieval success ratio (CRSR)	99.9%	PER-CRE-001	100%. All content was retrieved successfully
Content resolution signalling overhead, expressed by number of traversed ASes.	No specific reference value is defined because this metric will be used to compare approaches.	Covered inWP5	
Number of content records stored on CRE	A single CRE with 1GB HDD would store around 1.26 million content records (assuming the simplest content records)	PER-CRE-007	It was illustrated that at least 1 million content records could be stored.
Maximum number of users connected to the CP	No specific reference value is defined because this metric will be used to compare CP with HTTP server.	PER-CER-006	Tested with 5 clients establishing connections The number of HTTP connections was limited by the processing resources of Server used. Had collision issues.
Maximum publication rate (expressed in [pub/s])	Estimated publication time is around 10-30ms, hence the maximum publication rate would be around 30 - 100 pub/s.	Covered in PER-CRE-006 and PER-CRE-007	From a single source at 1 req at a time the average response time 0.1865 sec with a 100% successful registrations measured at the Client Side. The actual publication time at CRE was under 10ms. At 4 pub/s the success rate of 88.25%
Maximum query rate to root CRE (expressed in [req/s])	The estimated query rate should be around 500 - 1000 req/s per CRE.	PER-CRE-002-005	It was realised that the root CRE could handle query rates of up to 1000 req/s.
CRE response time (expressed by 95 percentile)	The estimated query time should be around 1-2 ms + CP processing time + RTT.	PER-CRE-007a-007f	Login: 187ms; Form 1: 68ms; Form 2: 55ms; Form 3 (Publication): 186ms;
Maximum query rate of SNME (expressed in [req/s])	No specific reference value was defined This metric will be evaluated in the testbed to provide data for scalability studies.	PER-SNME-008 PER-SNME-009	10 queries/s according to a Poisson distribution

Metric name	Reference value	TestID	Validated (comments)
SNME response time (expressed by 95 percentile)	No specific reference value was defined This metric will be evaluated in the testbed to assess the CRT time.	PER-SNME-008 PER-SNME-009	For a 10 queries/s rate the response time 95 % of the queries are below 0,2 s
Number of CS server by a SIC-SNME	A single SIC module is expected to handle dozens of CS.	PER-SNME-001 to 003	Optimal CS allocation is around 300 CS for a single CS
Maximum request rate of CME (expressed in [req/s])	No specific reference value was defined. This metric will be evaluated in the testbed to provide data for scalability studies.	PER-CME-002	Under 5 req/s the CME responded close to the average time. After 10 req/s the CPU becomes heavily consumed and the response time increases to undesired delays.
CME response time (expressed by 95 percentile)	No specific reference value was defined. This metric will be evaluated in the testbed to assess the CRT time.	PER-CME-002	The average CME response time was around 100ms.
Routing Convergence Time (RCT)	No specific reference value is defined, because this metric will be used to compare RAE with BGP.	PER-RAE-002 and 003	YES, The RAE convergence is in the order of BGP-4.
Number of stored network prefixes	RAE should be able to handle the same number of prefixes as in the current Internet, i.e. about 300 000	PER-RAE-001	YES, RAE was able to handle 300 000 prefixes
Lossless throughput of CAFE	No specific reference value is defined, because this metric will be used to compare CAFE with software IP router.	PER-CAFE-001 to PER-CAFE-004	YES, The CAFE throuput was the same as IPv6 router
Number of simultaneous flows	At 1 Gbps port, the maximum number of simultaneous flows should be about 10^5.	PER-CAFE-004	YES, CAFE was able to handle large number of flows without throughput degradation
Edge CAFE configuration latency	No specific reference value is defined, because this metric will be used to compare CAFE with software IP router.	PER-CAFE-005	YES, The configuration latency was about 46 ms.

Metric name	Reference value	TestID	Validated (comments)
Size of Forwarding Information Base (FIB)	No specific reference value is defined, because this metric will be used to compare FIB in CAFE and IP routers.	Covered inWP5	
Size of COMET header	The overhead introduced by COMET should not exceed the order of IPv6 header (40B) for the Internet scale network.	Covered inWP5	
Hop count	After route optimisation, the hop count should not exceed 7–8 hops.	Covered in WP5	
Bandwidth consumption	No specific reference value is defined, because this metric will be used to compare point-to-multipoint and unicast connections.	FUNC-UC1-019 FUNC-UC1-020 Additional analysis will be covered in WP5	

Table 33: Mapping of Quantitative Requirements to test cases

5.3 Evaluation Conclusions

Overall the functional and performance evaluation prove that the COMET architecture is feasible and deployable with certain performance limitations caused either by software development due bottlenecks that could be improved and are not architecture specific or protocol constraints.

Tested from a Client Side the CRE registrations from a single source at 1 req at a time the success rate was 100%. This was tested for 1 Million Registration attempts with 100% success. The CRE could resolve up to 1000 queries per second at 100% success rate both by root and authoritative CREs. Moreover, it was identified that the number of HTTP connections were limited by the server resources of the CP and CRE. The response times obtained for CRE registrations were also satisfactory considering that they incorporated the CP processing time, CRE processing time and the round trip times, at a time of around 50-200ms per content publication form (measured at the CC site). It was identified that at 4 pub/s the publication time was 260ms with a success rate of 88.25% which was caused by the request treatment at the CP of forwarding on request at a time to the CRE. For this to be improved changes to request handling at the CP are required.

For the CME, under 5 req/s the CME responded close to the average time caused by a single request at about 100ms, meeting the target. After 10 req/s however the CPU becomes heavily consumed and the response time increases to undesired delays. This is a scalability constraint and should be considered in future CME versions. For the RAE-CME interface the CPU and Memory at the CME managed to handle an average query rate of about 500 queries/second and 600 queries/second. It has to be noted that this “maximum” limits were influenced by the TCP protocol established between the RAE and CME entities. Even when the query rate was increased in the order of 1K, 10K and 100K queries per second the TCP protocol constraints the rate to ensure 100% success.

Regarding the number of CS served by a SIC-SNME, even though just for memory/CPU considerations it was shown that a single SIC module could easily manage at least 500 CS, taking into account query rate/response time a figure of 350 CS (70%) is more realistic. With regards to the maximum query rate & SNME response time, using the 70% occupation for a single SIC, the SNME is able to manage query rates of 10 queries/s (according to a Poisson distribution) obtaining response times whose 95% percentile is around 0,2 ms. This value is 1 order of magnitude below the target value for global response time for the entire system, estimated in 2,5 s.

As far as the RAE performance is concerned, the performed tests confirmed that RAE prototype is able to handle large number of prefixes. In the analysed cases, the RAE running on the ordinary PC machine was able to handle 500 000 prefixes (currently, there are 300 000 prefixes in the Internet). The routing convergence time measured for basic stressing events, i.e. prefix advertisement/withdrawal, link failure/repair is at similar level as BGP-4 convergence times reported in [12]. During the experiment, the CPU was recognised as the main bottleneck. Therefore, the RAE prototype should be optimised from the CPU usage point of view. Moreover, we recommend deploying RAE on machines with powerful processor to reduce the RCT.

For the CAFE performance, the throughput of CAFE is similar to the throughput of IPv6 software router running on the same hardware. Moreover, the edge CAFE is able to forward packets from 10000 simultaneously running flows. The edge CAFE configuration time is about 46 ms, which is relativelly small values comparing to tolerable resolution time.

6 COMET-ENVISION

This section explains the steps carried out in order to deploy and test a COMET ENVISION joint scenario that can demonstrate how the results of one project are useful for the other and vice-versa.

Basically, ENVISION, by using the CINA system, can provide cost information to external entities about links connecting two neighbouring ISP, discriminating between both directions in the link. Since discovery and set-up of optimal paths for content download between CCs and CSs in different ISP is one of COMET's main features, this cost information can be used for refining the path decision algorithm implemented in every COMET CME.

When a CC requests a content, The ISP CME requests from CINA the cost of the incoming links from neighbouring ISP, but only for those included in the list of candidate paths for content retrieval. This cost information can then be fed in the CME decision algorithm, together with IPTD, IPLR, server Load, Bandwidth and Path Length, modifying the results of the server selection.

6.1 *COMET-ENVISION Layout*

The following figure illustrates the basic Layout of COMET-ENVISION joint scenario:

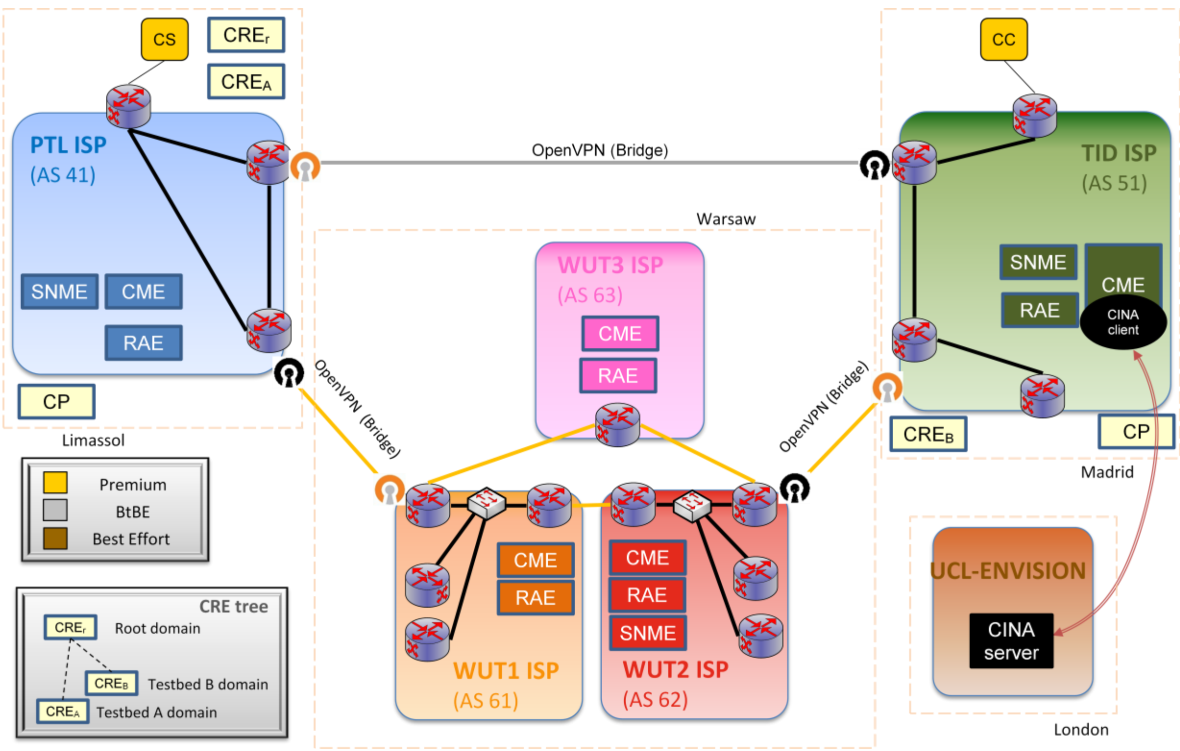


Figure 111: COMET-ENVISION Layout

A CINA server is deployed in a testbed located at UCL premises (London) which is accessible via Internet by using the following URL http://128.40.39.246/cgi-bin. TID local testbed in the COMET federated access has been configured to allow http access on the port 80 for that internet address

The CME has been enhanced with a CINA client that can request specific link costs to the CINA. This behaviour can be activated/deactivated through the CME Web Interface, thus enabling to compare the behaviour of the decision algorithm with CINA on and CINA off.

For demonstration purposes, the CINA enhanced CME has been only deployed at TID, so tests will involve a CC at TID requesting contents from a CSs located at other ISPs. Therefore, the cost information retrieved by CINA involves only the following links and traffic directions:

- From AS 41 to 51 (contents transmitted from PTL to TID) with an assigned cost of 12.5
- From AS 62 to 53 (contents transmitted from WUT to TID), with an assigned cost of 33

In this context, a greater value means a higher cost, so in general lines paths using the AS62-51 link will be assigned a worse ranking (worse = lower value) by the decision algorithm than those using the link AS41-51. Actual ranking can vary according to the configuration parameters of the decision algorithm, especially the aspiration level, which defines the importance of a parameter in the computation of the path ranking.

6.2 COMET-ENVISION Use Case

To demonstrate how COMET and ENVISION can interwork, a use case has been defined to show how the decision algorithm can be swayed when the CINA is activated in CME and consequently the cost information for incoming links is fed into the decision algorithm.

For this use case, a PR CC is deployed at TID and a PR CS at PTL. The CS information is stored in a Content Record at TID, the QoS parameters for the Content Source associated to the CS being equal to those defined in Section 3.7.1.1 for the PR CSs used in Use Case 1. Therefore, when the Content Name is requested and CINA is deactivated, the path chosen by the decision algorithm will be the same than in COMET Use Case 1, the long one crossing WUT ISPs.

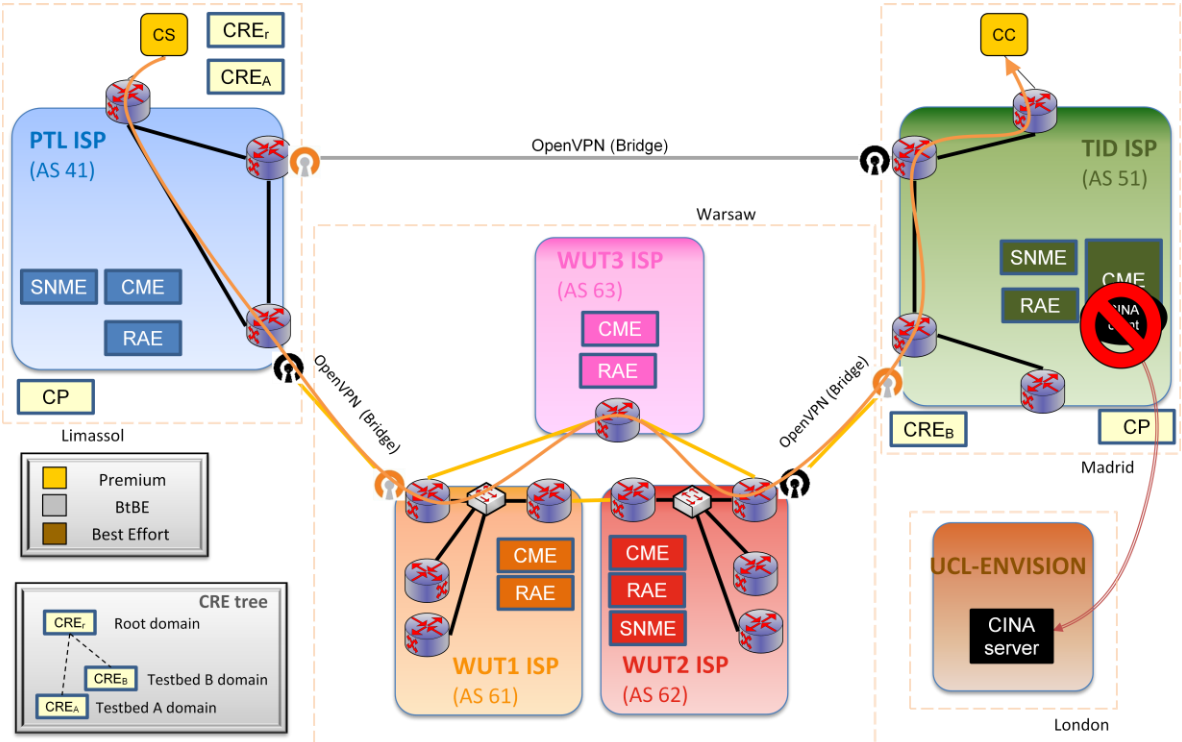


Figure 112: Behaviour without CINA Activated

To demonstrate the ENVISION-COMET interworking capabilities, CINA will be activated in TID CME and the following parameters will be set for the cost parameters in the decision algorithm

- Cost will set to strict mode, so it will reject any path not matching the QoS levels.
- Reservation Level will be set to 60, so the cost defined in the CINA server will be taken into consideration (maximum cost is 30 for TID links)
- The Aspiration Level will be set to 0.01, thus maximizing the contribution of this parameter to the ranking estimation.

So, when the Content Name is requested, COMET will retrieve the cost for the incoming links to TID (AS41 → AS51 and AS62 → AS51) and select the direct link between PTL and TID as the optimal option as illustrated in the next figure:

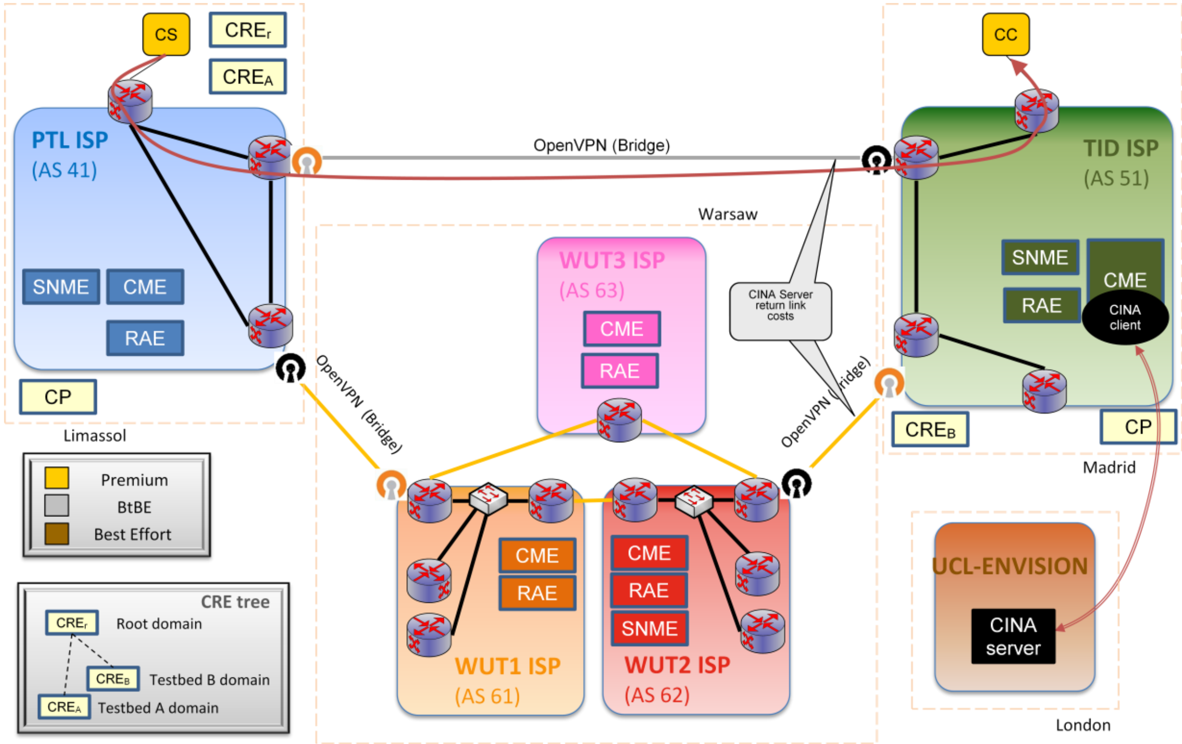


Figure 113: Behaviour with CINA Activated

Therefore, this use case demonstrates how the inclusion in COMET decision algorithm of the link costs provided by ENVISION can modify the final selected path.

6.3 *COMET-ENVISION Interworking*

For the purpose of the COMET-ENVISION use case, an updated version of CME software was implemented with the following modifications:

- Integration of CINA client software to Path Manager component of CME, which requests from CINA server the CINA cost per incoming link,
- Extension of CME administration interface, to activate the CINA client, configure CINA server's URI, and CINA cost's decision algorithm mode, reservation and aspiration levels,
- Extension of decision algorithm to take into account the received CINA cost (if CINA client is activated, otherwise ignored).

6.4 *COMET-ENVISION Functional Tests*

Based on the Use Case defined in Section 3 the following pieces of information have been collected to demonstrate the behaviour of the CME decision algorithm with CINA off or CINA on.

First, it is important to have in mind that there are three posible paths in the COMET federated testbed content can be downloaded from PTL to TID:

- AS 41 → AS 51 (Direct Link)
- AS41 → AS 61 → AS 62 → AS 51 (WUT Path)
- AS41 → AS 61 → AS 63 → AS 62 → AS 51 (WUT Path

With CINA deactivated AS41/ AS51 will obtain the worst ranking (0.90707064) compared to the other two that will assigned a value of (0.9127703). AS41/AS 61/AS 62/AS 51 is eventually selected because it has fewer hops than AS41/AS 61/AS63/AS 62/AS 51. The log from the decision process with the rank assignation is shown in the next figure.

```
2012-11-21 16:45:11,786 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 79.999695, Infinity, 96.9
9981, 0.90707064, 1.010101
2012-11-21 16:45:11,786 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 51] rate=0.90707064
2012-11-21 16:45:11,786 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 49.99981, Infinity, 72.49
9855, 0.9979799, 1.010101
2012-11-21 16:45:11,786 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 61, 63, 62, 51] rate=0.91277903
2012-11-21 16:45:11,786 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 59.99977, Infinity, 73.49
985, 1.0020202, 1.010101
2012-11-21 16:45:11,786 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 61, 62, 51] rate=0.91277903
2012-11-21 16:45:11,786 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.91277903
2012-11-21 16:45:11,786 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.91277903
2012-11-21 16:45:11,787 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.90707064
```

Figure 114: Ranking Assignment with CINA deactivated

By using the CAFE visualisator tool available in COMET, it can also be shown that the traffic is going through CAFES in TID (TID link to PTL, TID link to WUT) and entering WUT through the border CAFE in AS 62 (WUT Link to TID), as depicted in the following figure

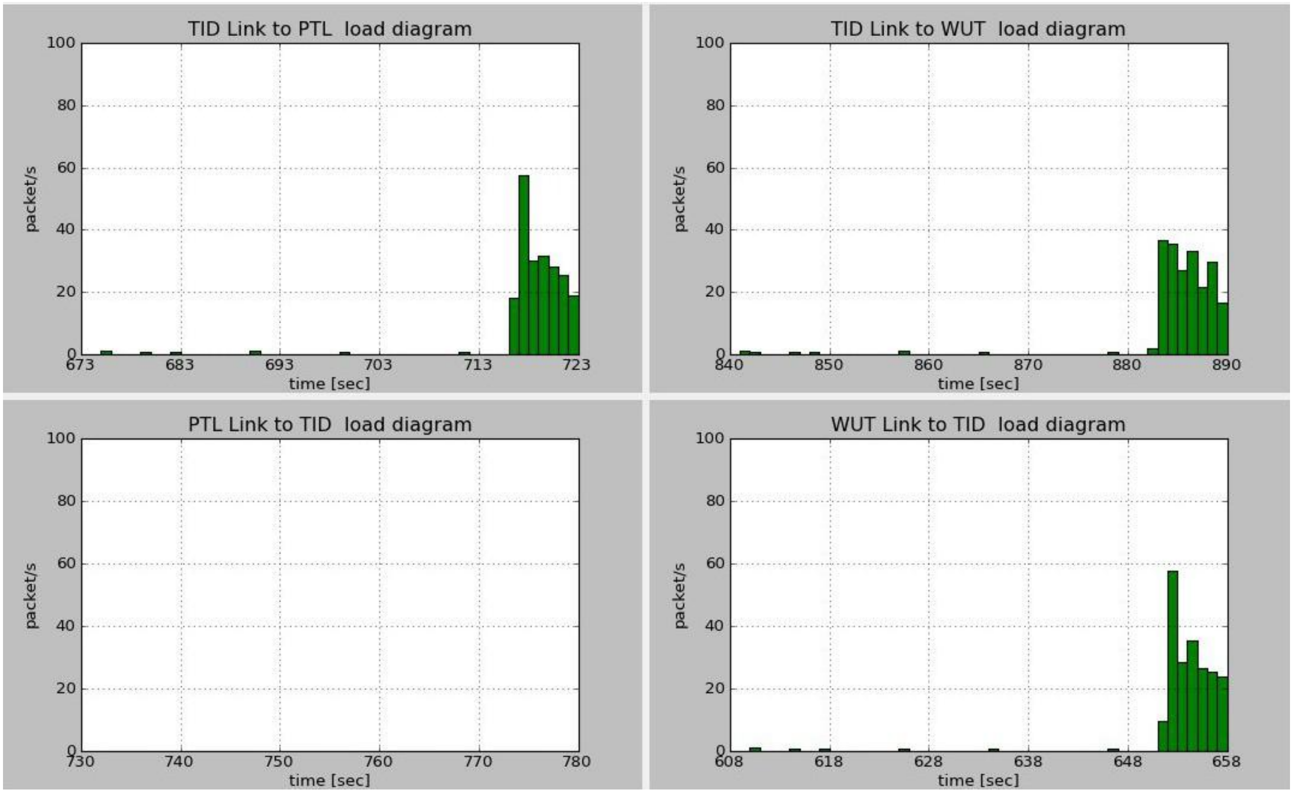


Figure 115: Traffic in CAFEs with CINA deactivated

When CINA is activated and content is retrieved, the cost for links entering TID from PTL and WUT is retrieved from the CINA server at UCL premises (URL isURL http://128.40.39.246/cgi-bin) . This retrieval operation is shown in the following extract of the CME log.

```
2012-11-21 16:48:38,169 DEBUG PathDiscoveryImpl - Initializing CINA client with URI: http://128.40.39.246/cgi-bin.
2012-11-21 16:48:38,198 DEBUG header - >> "GET /cgi-bin/NetworkMap.cgi HTTP/1.1[\r][\n]"
2012-11-21 16:48:38,198 DEBUG header - >> "ACCEPT: application/cina-networkmap+json,application/cina-error+json[\r][\n]"
2012-11-21 16:48:38,198 DEBUG header - >> "User-Agent: Jakarta Commons-HttpClient/3.1[\r][\n]"
2012-11-21 16:48:38,198 DEBUG header - >> "Host: 128.40.39.246[\r][\n]"
2012-11-21 16:48:38,198 DEBUG header - >> "[\r][\n]"
2012-11-21 16:48:38,328 DEBUG header - << "HTTP/1.1 200 OK[\r][\n]"
2012-11-21 16:48:38,328 DEBUG header - << "HTTP/1.1 200 OK[\r][\n]"
2012-11-21 16:48:38,328 DEBUG header - << "Content-type: application/cina-networkmap+json[\r][\n]"
2012-11-21 16:48:38,328 DEBUG header - << "Content-length: 959[\r][\n]"
2012-11-21 16:48:38,329 DEBUG header - << "Date: Wed, 21 Nov 2012 13:25:56 GMT[\r][\n]"
2012-11-21 16:48:38,329 DEBUG header - << "Server: lighttpd/1.4.28[\r][\n]"
2012-11-21 16:48:38,329 DEBUG header - << "[\r][\n]"
2012-11-21 16:48:38,355 DEBUG content - << "{"data":{"map-vtag":"123456780","map":{"ASTID":{"ipv6":["2A02:9008:0:1911:0:50:56
A3:4F","2A02:9008:0:1919:0:50:56A3:5F","2A02:9008:0:1918:0:50:56A3:5D"],"ipv4":["10.95.51.3","10.95.51.81","10.95.51.85"]},"A
SPT":{"ipv6":["3105::6","2001:7f8:4::8a61:1","2001:7f8:4::3f89:2","2001:7f8:4::1a00:1"],"ipv4":["10.50.50.3","10.50.50.200","
10.50.50.240","10.50.50.250"]},"ASWUT2":{"ipv6":["2001:67c:24cc:31d1:0:0:ae:2","2001:67c:24cc:31d0::cafe:c","2001:67c:24cc:31
d0::cafe:a","2001:67c:24cc:31d0::cafe:a"],"ipv4":["10.203.62.228","10.203.61.247","10.203.62.246","10.203.62.249"]},"ASWUT3":
{"ipv6":["2001:67c:24cc:31e1:0:0:ae:3","2001:67c:24cc:31e0::cafe:9","2001:67c:24cc:31e0::cafe:9"],"ipv4":["10.203.63.228","10
.203.63.246","10.203.62.247"]},"ASWUT1":{"ipv6":["2001:67c:24cc:31c1:0:0:ae:1","2001:67c:24cc:31c0::cafe:f","2001:67c:24cc:31
c0::cafe:d","2001:67c:24cc:31c0::cafe:d"],"ipv4":["10.203.61.228","10.203.61.246","10.203.63.247","10.203.61.250"]}}},"meta":
{}}"
```

Figure 116: Cost MAP retrieved from CINA Server

So the paths will be complemented with the following cost information

- AS 41 → AS 51 (Direct Link). Cost 12.5
- AS41 → AS 61 → AS 62 → AS 51 (WUT Path). Cost 30
- AS41 → AS 61 → AS 63 → AS 62 → AS 51 (WUT Path). Cost 30.

In this case, the ranking assigned to AS41/AS51 is 0.799633 as opposed to 0.45454544 for the other two paths. Therefore, the selected path with CINA activated is the direct one between PTL and TID. The CME log from the decision process with the rank assignation is shown in the next figure.

```
2012-11-21 16:48:39,258 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 79.999695, Infinity, 96.9
9981, 0.90707064, 0.7996633
2012-11-21 16:48:39,258 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 51] rate=0.7996633
2012-11-21 16:48:39,258 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 59.99977, Infinity, 73.49
985, 1.0020202, 0.45454544
2012-11-21 16:48:39,258 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 61, 62, 51] rate=0.45454544
2012-11-21 16:48:39,258 DEBUG DecisionMakerImpl - sload, plength, bw, iptd, iplr, cina: 0.91277903, 49.99981, Infinity, 72.49
9855, 0.9979799, 0.45454544
2012-11-21 16:48:39,259 DEBUG DecisionMakerImpl - Server 3101:0:0:0:0:0:0:3 and path [41, 61, 63, 62, 51] rate=0.45454544
2012-11-21 16:48:39,259 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.7996633
2012-11-21 16:48:39,259 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.45454544
2012-11-21 16:48:39,259 DEBUG DecisionMakerUtil - Server 3101:0:0:0:0:0:0:3 rate=0.45454544
```

Figure 117: Ranking Assignment with CINA activated

By using the CAFE visualisator tool available in COMET, it can also be shown that the traffic is going through CAFEs in TID (TID link to PTL only) and entering PTL through the border CAFE in AS 41 (PTL link to TID), as depicted in the following figure.

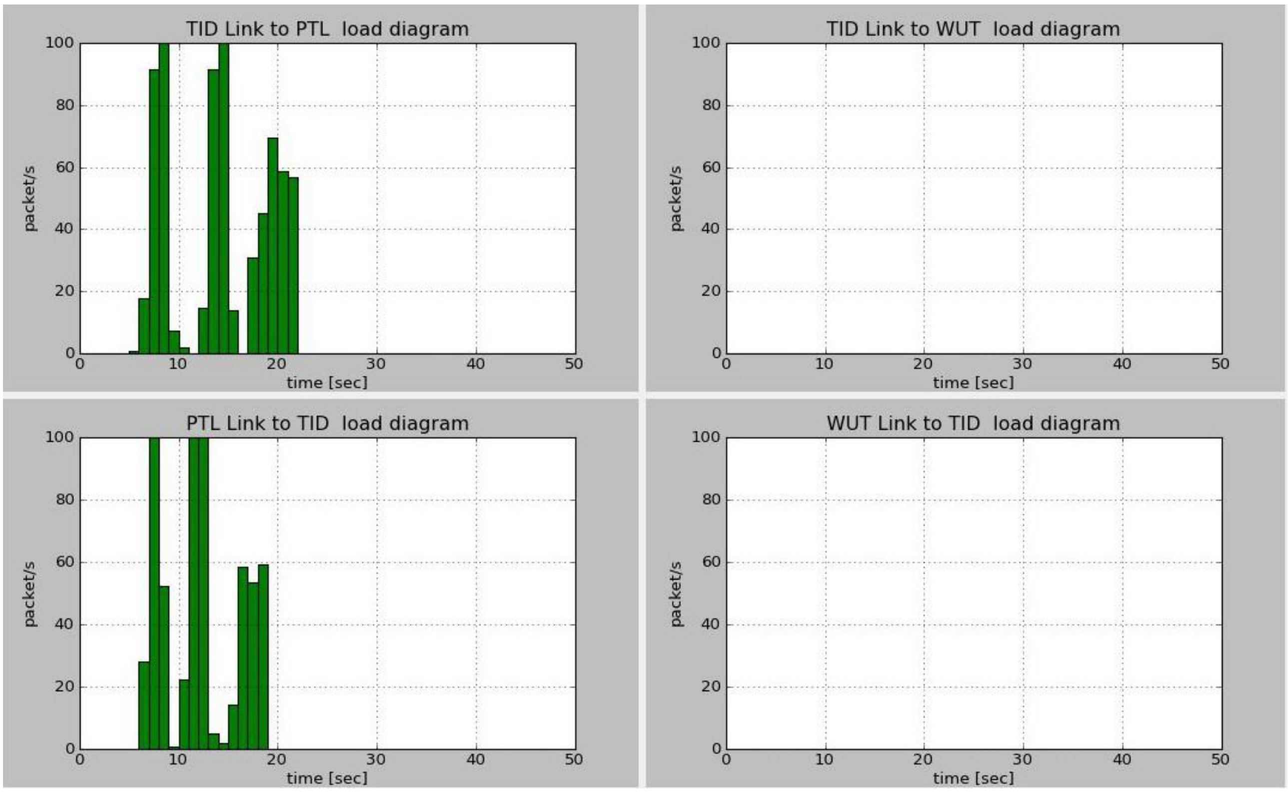


Figure 118: Traffic in CAFEs with CINA activated

Thus, it is shown how the path selection varies according to CINA being activated or not.

7 Summary and Conclusions

D6.2 has described the successful demonstration of the COMET system targeting the decoupled proposal, which allows easier deployment. The federated testbed has been described thoroughly in section 3 with references to more detailed configurations in Appendix (Annex) A. Moreover, an extensive evaluation has been performed both on functionality (operability) as well as on performance with emphasis on scalability.

The description of Section 3 together with the associated Annex can be considered as a specification blueprints for future work, both in utilising the COMET architecture beyond the length of the project by the involved companies and for future research by the consortium members. The selected use cases illustrated later that they where wisely chosen for functional testing.

The functional testing successfully demonstrated:

- Routing awareness.
- Content publication
- Content request.
- Name resolution.
- Successful demonstration of obtaining the load of the sources at the source.
- How path/server decision process works effectively.
- Successful demonstration of path configuration process.
- Successful content delivery.
- How the system reacts effectively when its high in terms of load.
- How the system successfully reacts when a new server is added to the architecture.
- How the system works when a new user asks for the Content and the new server added is taken into account.

Moreover, it was shown:

- How a Content Owner can change the type of CSs distribution of the Content.
- How VoD Content consumption works effectively using the same Content Record.
- Successful demonstration of how CCs in different domains can access to the same Content Name.
- Successful demonstration of how the CME is capable of distinguishing users of different CoS and assigning them to different servers for Content retrieval.
- How the CME does manage the CoS rules efficiently.
- How COMET can assign optimal paths according to the QoS/BW specified in the Content Record.
- Demonstration of how COMET will not assign paths whose quality is lower than those defined in the Content Record.
- Successful demonstration of how, with every other network parameter the same, the path selection is ruled by servers' load status.

Besides the aforementioned, two dedicated functional tests on point to multipoint streaming were demonstrated, showing how Content Streaming Relay (CSR) can support this capability and how CSR may reduce traffic carried on inter-domain links. Several functional tests related to the P2P Offloading case where successfully executed, namely, routing awareness, how a Content Owner distributes a Content publishing a Content Record with the number of Content Sources describing the characteristics and requirement of the streaming Servers, streaming distribution and finally efficient system reaction to high load on streaming servers.

An extensive number of performance tests were also performed to evaluate the different system components of COMET in terms of performance parameters that could be utilised for analysis and simulation by WP5 as well as using stress testers to evaluate scalability limitations based on actual implementation stress testing. The following components where hence tested for limitations: CRE, CME (Isolated), SNME, RAE, CAFE and CC. To achieve this, several stress testers where created emulating the behaviour of message requests by the architecture to stress test a particular component namely: CRE Feeder, CME Stress Tester, CP Stress Tester, CC Stress Tester, RAE Stress Tester, SMA stress tester and CAFE stress tester with different settings when testing the aforementioned components.

A large set of results were obtained illustrating the scalability limitations of the system which allow us to claim that with the appropriate settings the COMET system is deployable in the real internet environment between real ISPs. A number of performance tests were also obtained as benchmarks for WP5 scalability studies e.g. average response time. The functional validation tests illustrated that the COMET architecture is fully functional and feasible. The performance tests illustrated that the majority of the expected targets where met with some bottleneck limitations caused by software development issues and protocol congestion control procedure only.

Finally the COMET-ENVISION interworking solution together with the proposed use case used for evaluation verified that by using the CINA system further cost information can be provided to external entities about links connecting two neighbouring ISP, discriminating between both directions in the link. Since discovery and set-up of optimal paths for content download between CCs and CSs in different ISP is one of COMET's main features, this cost information was able to refine the path decision algorithm implemented in every COMET CME and successfully making a more informative path/server selection according to the additional criteria provided.

Annex A Federated Testbed Raw Data

A.1 TID LOCAL TESTBED

A.1.1 Addressing Schema

Machine	Interface	MAC Address	IPv4	IPv6
CRE-B	eth1	00:50:56:a3:00:4e	10.95.51.2	2a02:9008:0:1911:0:50:56a3:4e
cme-tid	eth1	00:50:56:a3:00:4f	10.95.51.3	2a02:9008:0:1911:0:50:56a3:4f
SNME-TID	eth1	00:50:56:a3:00:50	10.95.51.4	2a02:9008:0:1911:0:50:56a3:50
RAE-TID	eth1	00:50:56:a3:00:51	10.95.51.5	2a02:9008:0:1911:0:50:56a3:51
CAFE-TIDpt	eth1 eth2 eth3	00:50:56:a3:00:52 00:50:56:a3:00:5c 00:50:56:a3:00:5d	10.95.51.65 10.95.51.109 10.95.51.81	2a02:9008:0:1914:0:50:56a3:52 2a02:9008:0:1916:0:50:56a3:5c 2a02:9008:0:1918:0:50:56a3:5d
CAFE-TIDwut	br0 eth1 eth2 eth3	da:08:9a:71:9d:9f 00:50:56:a3:00:5e 00:50:56:a3:00:53 00:50:56:a3:00:5f	10.203.62.250 10.95.51.69 10.95.51.110 10.95.51.85	2a02:9008:0:1915:0:50:56a3:5e 2a02:9008:0:1916:0:50:56a3:53 2a02:9008:0:1919:0:50:56a3:5f
CAFE-TID2	br1 eth1 eth2 eth3	00:50:56:a3:00:54 00:50:56:a3:00:5a 00:50:56:a3:00:54 00:50:56:a3:00:5b	10.95.51.33 10.95.51.1 10.95.51.66	2a02:9008:0:1912:0:50:56a3:54 2a02:9008:0:1911:0:50:56a3:5a 2a02:9008:0:1912:0:50:56a3:54 2a02:9008:0:1914:0:50:56a3:5b
CAFE-TID3	eth1 eth2	00:50:56:a3:00:55 00:50:56:a3:00:60	10.95.51.49 10.95.51.70	2a02:9008:0:1913:0:50:56a3:55 2a02:9008:0:1915:0:50:56a3:60
CC-TID1	eth1	00:50:56:a3:00:62	10.95.51.37	2a02:9008:0:1912:0:50:56a3:62
CC-TID2	eth1	00:50:56:a3:00:45	10.95.51.34	2a02:9008:0:1912:0:50:56a3:45
CC-TID3	eth1	00:50:56:a3:00:61	10.95.51.50	2a02:9008:0:1913:0:50:56a3:61
CC-TID4	eth1	00:50:56:a3:00:02	10.95.51.52	2a02:9008:0:1913:0:50:56a3:64
CS-TID2	eth1	00:50:56:a3:00:48	10.95.51.35	2a02:9008:0:1912:0:50:56a3:48
CS-TID3	eth0	00:50:56:a3:00:05	10.95.51.51	2a02:9008:0:1913:0:50:56a3:62
CS-TID4	eth1	00:50:56:a3:00:07	10.95.51.53	2a02:9008:0:1913:0:50:56a3:63
cs-p2p	eth1	00:50:56:a3:00:4a	10.95.51.36	2a02:9008:0:1912:0:50:56a3:4a
cafe_B	eth2	08:00:27:6a:91:2b	10.203.62.249	

Table 34: TID Addressing Schema

A.1.2 Forwarding Keys

CAFE	Key Name	Interface (device)	Destination MAC Address	Remote cafe associated	Remote interface
CAFE-TID2	0xfa	eth2	00:50:56:a3:00:54	CAFE-TID2	eth2
CAFE-TID2	0xff	eth3	00:50:56:a3:00:52	CAFE-TIDpt	eth1
CAFE-TIDpt	0xfe	eth2	00:50:56:a3:00:53	CAFE-TIDwut	eth2
CAFE-TIDpt	0xff	eth1	00:50:56:a3:00:5b	CAFE-TID2	eth3
CAFE-TIDpt	0xfc	br1	16:24:e6:b0:76:c0	cafe-pttid	eth3
CAFE-TIDwut	0xfc	br0	08:00:27:6a:91:2b	cafe_a	eth2
CAFE-TIDwut	0xfd	eth1	00:50:56:a3:00:60	CAFE-TID3	eth2
CAFE-TIDwut	0xfe	eth2	00:50:56:a3:00:5c	CAFE-TIDpt	eth2
CAFE-TID3	0xfb	eth1	00:50:56:a3:00:55	CAFE-TID3	eth1
CAFE-TID3	0xfd	eth2	00:50:56:a3:00:5e	CAFE-TIDwut	eth1

Table 35: TID Forwarding Keys

A.1.3 IP Prefixes/CAFE Mapping

Prefix	Prefix Length	Edge Cafe
10.95.51.32	28	10.95.51.33
10.95.51.48	28	10.95.51.49

Table 36: TID Subnet/CAFE Mapping in IPv4

Prefix	Prefix Length	Edge Cafe
2a02:9008:0:1912:0:0:0:0	28	2a02:9008:0:1912:0:50:56a3:54
2a02:9008:0:1913:0:0:0:0	28	2a02:9008:0:1913:0:50:56a3:55

Table 37: TID Subnet/CAFE Mapping in IPv4

A.1.4 Path Configuration in RAE

AS/Range Source	AS/Range sink	COS	BW	IPLR	IPTD
51	62	BtBE	10000000	0.000005	0.005
51	62	PR	20000000	0.000001	0.001
51	41	BtBE	10000000	0.000008	0.010
51	41	PR	16000000	0.000002	0.002
2a02:9008:0:1912::	2a02:9008:0:1913::	BtBE	20000000	0.000001	0.001
2a02:9008:0:1912::	2a02:9008:0:1913::	PR	20000000	0.000001	0.001
2a02:9008:0:1913::	2a02:9008:0:1912::	BtBE	20000000	0.000001	0.0005
2a02:9008:0:1913::	2a02:9008:0:1912::	PR	20000000	0.000001	0.001
2a02:9008:0:1912::	62	BtBE	20000000	0.000001	0.0005
2a02:9008:0:1912::	62	PR	20000000	0.000001	0.001
2a02:9008:0:1912::	41	BtBE	20000000	0.000001	0.0005
2a02:9008:0:1912::	41	PR	20000000	0.000001	0.001
2a02:9008:0:1913::	62	BtBE	20000000	0.000001	0.0005
2a02:9008:0:1913::	62	PR	20000000	0.000001	0.001
2a02:9008:0:1913::	41	BtBE	20000000	0.000001	0.0005
2a02:9008:0:1913::	41	PR	20000000	0.000001	0.001
62	2a02:9008:0:1912::	BtBE	20000000	0.000001	0.0005
62	2a02:9008:0:1912::	PR	20000000	0.000001	0.001
41	2a02:9008:0:1912::	BtBE	20000000	0.000001	0.0005
41	2a02:9008:0:1912::	PR	20000000	0.000001	0.001
62	2a02:9008:0:1913::	BtBE	20000000	0.000001	0.0005

AS/Range Source	AS/Range sink	COS	BW	IPLR	IPTD
62	2a02:9008:0:1913::	PR	20000000	0.000001	0.001
41	2a02:9008:0:1913::	BtBE	20000000	0.000001	0.0005
41	2a02:9008:0:1913::	PR	20000000	0.000001	0.001
41	62	BtBE	20000000	0.000001	0.0005
41	62	PR	20000000	0.000001	0.001
62	41	BtBE	20000000	0.000001	0.0005
62	41	PR	20000000	0.000001	0.001
2a02:9008:0:1912::	2a02:9008:0:1913::	BtBE	50000000	0.0000001	0.0005
2a02:9008:0:1912::	2a02:9008:0:1913::	PR	50000000	0.0000001	0.0005
2a02:9008:0:1913::	2a02:9008:0:1912::	BtBE	50000000	0.0000001	0.0005
2a02:9008:0:1912::	2a02:9008:0:1912::	PR	50000000	0.0000001	0.0005

Table 38: TID Path Configuration in RAE

A.1.5 Peering AS/CAFE Mappings

AS Path	CoS	Source CAFE	Sink CAFE	Peer CME	BW
51,62	PR	10.95.51.85	10.203.62.227	10.203.62.228	100MB
51,62	BTBE	10.95.51.85	10.203.62.227	10.203.62.228	50MB
62,51	PR	10.203.62.227	10.95.51.85	10.203.62.228	100MB
62,51	BTBE	10.203.62.227	10.95.51.85	10.203.62.228	50MB
51,41	PR	10.95.51.81	10.50.50.3	10.50.50.6	100MB
51,41	BTBE	10.95.51.81	10.50.50.3	10.50.50.6	50MB
41,51	PR	10.50.50.3	10.95.51.81	10.50.50.6	100MB
41,51	BTBE	10.50.50.3	10.95.51.81	10.50.50.6	50MB

Table 39: TID Peering AS/CAFE Mappings for IPv4

AS Path	CoS	Source CAFE	Sink CAFE	Peer CME	BW
51,62	PR	2a02:9008:0:1919:0:50:56a3:5f	2001:67c:24cc:31d0::cafe:a	2001:67c:24cc:31d1::ae:2	100MB
51,62	BTBE	2a02:9008:0:1919:0:50:56a3:5f	2001:67c:24cc:31d0::cafe:a	2001:67c:24cc:31d1::ae:2	50MB
62,51	PR	2001:67c:24cc:31d0::cafe:a	2a02:9008:0:1919:0:50:56a3:5f	2001:67c:24cc:31d1::ae:2	100MB
62,51	BTBE	2001:67c:24cc:31d0::cafe:a	2a02:9008:0:1919:0:50:56a3:5f	2001:67c:24cc:31d1::ae:2	50MB
51,41	PR	2a02:9008:0:1918:0:50:56a3:5d	3105::3	3105::6	100MB
51,41	BTBE	2a02:9008:0:1918:0:50:56a3:5d	3105::3	3105::6	50MB
41,51	PR	3105::3	2a02:9008:0:1918:0:50:56a3:5d	3105::6	100MB
41,51	BTBE	3105::3	2a02:9008:0:1918:0:50:56a3:5d	3105::6	50MB

Table 40: TID Peering AS/CAFE Mappings for IPv6

A.1.6 Peering CAFEs/Keys Mappings

Type	$CAFE_{or} \rightarrow CAFE_{dest}$	Key list (reverse order)	CoS	BW
TRANSIT	10.95.51.33 → 10.95.51.33	0xfa	PR	1GB
TRANSIT	10.95.51.33 → 10.95.51.49	0xfd,0xfe,0xff	PR	1GB
TRANSIT	10.95.51.49 → 10.95.51.33	0xff,0xfe,0xfd	PR	1GB
TRANSIT	10.95.51.49 → 10.95.51.49	0xfb	PR	1GB
TRANSIT	10.95.51.33 → 10.95.51.85	0xfe,0xff	PR	1GB
TRANSIT	10.95.51.85 → 10.95.51.33	0xff,0xfe	PR	1GB
TRANSIT	10.95.51.49 → 10.95.51.85	0xfd	PR	1GB
TRANSIT	10.95.51.85 → 10.95.51.49	0xfd	PR	1GB
TRANSIT	10.95.51.33 → 10.95.51.81	0xff	PR	1GB
TRANSIT	10.95.51.49 → 10.95.51.81	0xfe,0xfd	PR	1GB
TRANSIT	10.95.51.81 → 10.95.51.33	0xff	PR	1GB
TRANSIT	10.95.51.81 → 10.95.51.49	0xfd,0xfe	PR	1GB
TRANSIT	10.95.51.33 → 10.95.51.33	0xfa	BtBE	500M
TRANSIT	10.95.51.33 → 10.95.51.49	0xfd,0xfe,0xff	BtBE	500M
TRANSIT	10.95.51.49 → 10.95.51.33	0xff,0xfe,0xfd	BtBE	500M
TRANSIT	10.95.51.49 → 10.95.51.49	0xfb	BtBE	500M
TRANSIT	10.95.51.33 → 10.95.51.85	0xfe,0xff	BtBE	500M
TRANSIT	10.95.51.85 → 10.95.51.33	0xff,0xfe	BtBE	500M
TRANSIT	10.95.51.49 → 10.95.51.85	0xfd	BtBE	500M
TRANSIT	10.95.51.85 → 10.95.51.49	0xfd	BtBE	500M
TRANSIT	10.95.51.33 → 10.95.51.81	0xff	BtBE	500M

Type	$CAFE_{or} \rightarrow CAFE_{dest}$	Key list (reverse order)	CoS	BW
TRANSIT	10.95.51.49 → 10.95.51.81	0xfe,0xfd	BtBE	500M
TRANSIT	10.95.51.81 → 10.95.51.33	0xff	BtBE	500M
TRANSIT	10.95.51.81 → 10.95.51.49	0xfd,0xfe	BtBE	500M
PEERING	10.203.62.227 →10.95.51.85	0xfc	PR	100MB
PEERING	10.95.51.81 → 10.50.50.3	0xfc	PR	100MB
PEERING	10.50.50.3 → 10.95.51.81	0xfc	PR	100MB
PEERING	10.95.51.85 → 10.203.62.227	0xfc	BTBE	50MB
PEERING	10.203.62.227 → 10.95.51.85	0xfc	BTBE	50MB
PEERING	10.95.51.81 → 10.50.50.3	0xfc	BTBE	50MB

Table 41: TIDPeering CAFEs/Keys Mappings in IPv4

Type	$CAFE_{or} \rightarrow CAFE_{dest}$	Key list (reverse order)	CoS	BW
TRANSIT	2a02:9008:0:1912:0:50:56a3:54 →2a02:9008:0:1912:0:50:56a3:54	0xfa	PR	1GB
TRANSIT	2a02:9008:0:1912:0:50:56a3:54 → 2a02:9008:0:1913:0:50:56a3:55	0xfd,0xfe,0xff	PR	1GB
TRANSIT	2a02:9008:0:1913:0:50:56a3:55 → 2a02:9008:0:1912:0:50:56a3:54	0xff,0xfe,0xfd	PR	1GB
TRANSIT	2a02:9008:0:1913:0:50:56a3:55 →2a02:9008:0:1913:0:50:56a3:55	0xfb	PR	1GB
TRANSIT	2a02:9008:0:1912:0:50:56a3:54 → 2a02:9008:0:1919:0:50:56a3:5f	0xfe,0xff	PR	1GB
TRANSIT	2a02:9008:0:1919:0:50:56a3:5f → 2a02:9008:0:1912:0:50:56a3:54	0xff,0xfe	PR	1GB
TRANSIT	2a02:9008:0:1913:0:50:56a3:55 → 2a02:9008:0:1919:0:50:56a3:5f	0xfd	PR	1GB
TRANSIT	2a02:9008:0:1919:0:50:56a3:5f → 2a02:9008:0:1913:0:50:56a3:55	0xfd	PR	1GB
TRANSIT	2a02:9008:0:1912:0:50:56a3:54 → 2a02:9008:0:1918:0:50:56a3:5d	0xff	PR	1GB
TRANSIT	2a02:9008:0:1918:0:50:56a3:5d → 2a02:9008:0:1912:0:50:56a3:54	0xff	PR	1GB
TRANSIT	2a02:9008:0:1913:0:50:56a3:55 → 2a02:9008:0:1918:0:50:56a3:5d	0xfe, 0xfd	PR	1GB

Type	$CAFE_{or}$ → $CAFE_{dest}$	Key list (reverse order)	CoS	BW
TRANSIT	2a02:9008:0:1918:0:50:56a3:5d → 2a02:9008:0:1913:0:50:56a3:55	0xfd, 0xfe	PR	1GB
TRANSIT	2a02:9008:0:1918:0:50:56a3:5d → 2a02:9008:0:1919:0:50:56a3:5f	0xfe	PR	1GB
TRANSIT	2a02:9008:0:1919:0:50:56a3:5f → 2a02:9008:0:1918:0:50:56a3:5d	0xfe	PR	1GB
TRANSIT	2a02:9008:0:1912:0:50:56a3:54 →2a02:9008:0:1912:0:50:56a3:54	0xfa	BtBE	500M
TRANSIT	2a02:9008:0:1912:0:50:56a3:54 → 2a02:9008:0:1913:0:50:56a3:55	0xfd,0xfe,0xff	BtBE	500M
TRANSIT	2a02:9008:0:1913:0:50:56a3:55 → 2a02:9008:0:1912:0:50:56a3:54	0xff,0xfe,0xfd	BtBE	500M
TRANSIT	2a02:9008:0:1913:0:50:56a3:55 →2a02:9008:0:1913:0:50:56a3:55	0xfb	BtBE	500M
TRANSIT	2a02:9008:0:1912:0:50:56a3:54 → 2a02:9008:0:1919:0:50:56a3:5f	0xfe,0xff	BtBE	500M
TRANSIT	2a02:9008:0:1919:0:50:56a3:5f → 2a02:9008:0:1912:0:50:56a3:54	0xff,0xfe	BtBE	500M
TRANSIT	2a02:9008:0:1913:0:50:56a3:55 → 2a02:9008:0:1919:0:50:56a3:5f	0xfd	BtBE	500M
TRANSIT	2a02:9008:0:1919:0:50:56a3:5f → 2a02:9008:0:1913:0:50:56a3:55	0xfd	BtBE	500M
TRANSIT	2a02:9008:0:1912:0:50:56a3:54 → 2a02:9008:0:1918:0:50:56a3:5d	0xff	BtBE	500MB
TRANSIT	2a02:9008:0:1918:0:50:56a3:5d → 2a02:9008:0:1912:0:50:56a3:54	0xff	BtBE	500MB
TRANSIT	2a02:9008:0:1913:0:50:56a3:55 → 2a02:9008:0:1918:0:50:56a3:5d	0xfe, 0xfd	BtBE	500MB
TRANSIT	2a02:9008:0:1918:0:50:56a3:5d → 2a02:9008:0:1913:0:50:56a3:55	0xfd, 0xfe	BtBE	500MB
TRANSIT	2a02:9008:0:1918:0:50:56a3:5d → 2a02:9008:0:1919:0:50:56a3:5f	0xfe	BtBE	500MB
TRANSIT	2a02:9008:0:1919:0:50:56a3:5f → 2a02:9008:0:1918:0:50:56a3:5d	0xfe	BtBE	500MB
PEERING	2a02:9008:0:1919:0:50:56a3:5f → 2001:67c:24cc:31d0::cafe:a	0xfc	PR	100MB
PEERING	2a02:9008:0:1918:0:50:56a3:5d → 3105::3	0xfc	PR	100MB
PEERING	3105::3 → 2a02:9008:0:1918:0:50:56a3:5d	0xfc	PR	100MB
PEERING	2a02:9008:0:1919:0:50:56a3:5f → 2001:67c:24cc:31d0::cafe:a	0xfc	BTBE	50M
PEERING	2001:67c:24cc:31d0::cafe:a→ 2a02:9008:0:1919:0:50:56a3:5f	0xfc	BTBE	50M

Type	$CAFE_{or} \rightarrow CAFE_{dest}$	Key list (reverse order)	CoS	BW
PEERING	2a02:9008:0:1918:0:50:56a3:5d → 3105::3	0xfc	BTBE	50MB
PEERING	3105::3 → 2a02:9008:0:1918:0:50:56a3:5d	0xfc	BTBE	50MB

Table 42: TID Peering CAFEs/Keys Mappings in IPv6

A.1.7 Content Client to COMET CoS Mappings

Host	Client IP	CoS
CC-TID1	10.95.51.37	BE
CC-TID2	10.95.51.34	PR
CC-TID3	10.95.51.50	PR
CC-TID4	10.95.51.52	PR

Table 43: TID CC to CoS Mapping in IPv4

Host	Client IP	CoS
CC-TID1	2a02:9008:0:1912:0:50:56a3:62	BE
CC-TID2	2a02:9008:0:1912:0:50:56a3:45	PR
CC-TID3	2a02:9008:0:1913:0:50:56a3:61	PR
CC-TID4	2a02:9008:0:1913:0:50:56a3:64	BtBE

Table 44: TID CC to CoS Mapping in IPv6

A.1.8 Content Names

Name	CoS	QoS	URL	MIME	CME
tid1.es/cleoP2P	BE	BW:10 Delay: 0,6 Loss: 0.1	http://10.95.51.36:8080/media/cleoP2P.torrent	text/torrent	10.95.51.3
tid1.es/cleoSD	PR	BW:25 Delay: 0,6 Loss: 100	http://10.95.51.51:8008/cleo.ts http://10.95.51.35:8008/cleo.ts	stream/mpeg	10.95.51.3
tid1.es/cleoSD_BtBE	BtBE	BW:25 Delay: 0,6 Loss: 100	http://10.95.51.52:8008/cleo.ts http://10.95.51.35:8008/cleo.ts	stream/mpeg	10.95.51.3
tid1.es/cleoSD_BE	BE	BW:25 Delay: 0,6 Loss: 10	http://10.95.51.53:8008/cleo.ts	stream/mpeg	10.95.51.3
tid1.es/cleoVOD	PR	BW:25 Delay: 0,6 Loss: 100	http://10.95.51.51/videos/CLEOPATRA_DISC_1_480p.ts http://10.95.51.35/videos/CLEOPATRA_DISC_1_480p.ts	stream/mpeg	10.95.51.3
tid1.es/cleoVOD_BtBE	BtBE	BW:25 Delay: 0,6 Loss: 100	http://10.95.51.52/videos/CLEOPATRA_DISC_1_480p.ts http://10.95.51.35/videos/CLEOPATRA_DISC_1_480p.ts	stream/mpeg	10.95.51.3
tid1.es/cleoVOD_BE	BE	BW:25 Delay: 0,6 Loss: 10	http://10.95.51.53/videos/CLEOPATRA_DISC_1_480p.ts	stream/mpeg	10.95.51.3
tid1.es/cloeMIX	PR/BE	BW:25 Delay: 0,6 Loss: 10	PR http://10.95.51.51:8008/cleo.ts http://10.95.51.35:8008/cleo.ts BE http://10.95.51.36:8080/media/cleoP2P.torrent	stream/mpeg	10.95.51.3

Table 45: TID Content Names defined in TID for IPv4

Name	CoS	QOS	URL	MIME	CME
tid1.es/cleoP2P	BE	BW:10 Delay: 0,6 Loss: 0.1	http://[2a02:9008:0:1912:0:50:56a3:4a]/media/cleoP2P.torrent	text/torrent	2a02:9008:0:1911:0:50:56a3:4e
tid1.es/cleoSD	PR	BW:25 Delay: 0,6 Loss: 100	http://[2a02:9008:0:1913:0:50:56a3:62]:8008/cleo.ts http://[2a02:9008:0:1912:0:50:56a3:48]:8008/cleo.ts	stream/mpeg	2a02:9008:0:1911:0:50:56a3:4e
tid1.es/cleoSD_BtBE	PR	BW:25 Delay: 0,6 Loss: 100	http://[2a02:9008:0:1913:0:50:56a3:63]:8008/cleo.ts	stream/mpeg	2a02:9008:0:1911:0:50:56a3:4e
tid1.es/cleoSD_BE	BE	BW:25 Delay: 0,6 Loss: 10	http://[2a02:9008:0:1913:0:50:56a3:63]:8008/cleo.ts	stream/mpeg	2a02:9008:0:1911:0:50:56a3:4e

tid1.es/cleo VOD	PR	BW:25 Delay: 0,6 Loss: 100	http://[2a02:9008:0:1913:0:50:56a3:62]/videos/CLEOPATRA_DISC_1_480p.ts http://[2a02:9008:0:1912:0:50:56a3:48]/videos/CLEOPATRA_DISC_1_480p.ts	stream/mpeg	2a02:9008:0:1911:0:50:56a3:4e
tid1.es/cleo VOD_BtBE	PR	BW:25 Delay: 0,6 Loss: 100	http://[2a02:9008:0:1913:0:50:56a3:63]/videos/CLEOPATRA_DISC_1_480p.ts	stream/mpeg	2a02:9008:0:1911:0:50:56a3:4e
tid1.es/cleo VOD_BE	BE	BW:25 Delay: 0,6 Loss: 10	http://[2a02:9008:0:1913:0:50:56a3:63]/videos/CLEOPATRA_DISC_1_480p.ts	stream/mpeg	2a02:9008:0:1911:0:50:56a3:4e
tid1.es/cloe MIX	PR/BE	BW:25 Delay: 0,6 Loss: 10	PR http://[2a02:9008:0:1913:0:50:56a3:62]:8008/cleo.ts http://[2a02:9008:0:1912:0:50:56a3:48]:8008/cleo.ts BE http://[2a02:9008:0:1912:0:50:56a3:4a]/media/cleoP2P.torrent	PR: stream/mpeg BE: text/torrent	2a02:9008:0:1911:0:50:56a3:4e

Table 46: TID Content Names defined for IPv6

A.1.9 OpenVPN Configuration

A.1.9.1 Server-Side configuration

```
mode server
tls-server

port 2011 ## default openvpn port
proto udp

dev tap1
up "/etc/openvpn/up.sh br1 tap1 1500"
down "/etc/openvpn/down.sh br1 tap1"

persist-key
persist-tun

ca ca.crt
cert server.crt
key server.key # This file should be kept secret
dh dh1024.pem

cipher none
comp-lzo

#DHCP Information
ifconfig-pool-persist ipp.txt
server-bridge 192.168.16.1 255.255.255.0 192.168.16.5 192.168.16.12
#push "route 10.95.51.0 255.255.255.0"
#push "route 10.203.62.0 255.255.255.0"
#push "route 10.203.63.0 255.255.255.0"
#push "route 10.203.61.0 255.255.255.0"
max-clients 10 ## set this to the max number of clients that should be connected at a time

#log and security
user nobody
group nogroup
keepalive 10 120
status openvpn-status.log
verb 3
log openvpn.log
```

A.1.9.2 Client-Side configuration

```
client
dev tap0
lladdr 72:b5:ce:ce:fa:40
proto udp
remote 194.29.169.3 2001
nobind
#resolv-retry infinite
#persist-key
#persist-tun
ca ca.crt
```

```
cert tid.crt
key tid.key
#tls-authta.key 1
ns-cert-type server
cipher none
```

A.1.9.3 Remote-Access configuration

A.1.9.3.1 Server-side

```
mode server

port 2012
proto udp

dev tap1

persist-key
persist-tun

ca /home/comet/openvpn/keys/ca.crt
cert /home/comet/openvpn/keys/server.crt
dh /home/comet/openvpn/keys/dh1024.pem
key /home/comet/openvpn/keys/server.key

cipher none
comp-lzo

# IPV4
ifconfig-pool-persist ipp.txt
server 10.95.51.40 255.255.255.248
push "route 10.95.51.0 255.255.255.0"
push "route 10.203.62.0 255.255.255.0"
push "route 10.203.63.0 255.255.255.0"
push "route 10.203.61.0 255.255.255.0"
push "route 10.50.50.0 255.255.255.0"
push "route 101.101.101.0 255.255.255.0"

keepalive 10 120

status /home/comet/openvpn/openvpn-udp-status.log
log-append  /home/comet/openvpn/openvpn-udp.log

verb 5

mute 20
```

A.1.9.3.2 Client-side example

```
client

dev tap

proto udp

remote 130.206.214.70 2012

resolv-retry infinite

nobind
```

persist-key

persist-tun

ca ca.crt

cert nacho.crt

key nacho.key

cipher none

comp-lzo

verb 3

mute 2

A.1.10 Quagga Configuration

A.1.10.1 CAFE-TIDPT Configuration

- daemons

```
zebra=yes
bgpd=yes
ospfd=yes
ospf6d=yes
ripd=no
ripngd=no
isisd=no
```

- zebra.conf

```
comet@CAFE-TIDpt:/etc/quagga$ cat zebra.conf
hostname Router
password zebra
enable password zebra
interface br1
ip address 192.168.16.1/24
link-detect
ipv6 nd suppress-ra

interface eth1
link-detect
ipv6 nd suppress-ra

interface eth2
link-detect
ipv6 nd suppress-ra

interface eth3
link-detect
ipv6 nd suppress-ra

!ip route 0.0.0.0/0 203.181.89.241
```

```
ip forwarding
ipv6 forwarding
```

- bgp.conf

```
! BGP configurtion for COMET federated testbed
! border router in TID
! 2012/02/08 18:43:00
!
hostname TID-PT
password zebra
!
log file /var/log/quagga/bgpd.log
!
!debug bgp updates
!
router bgp 51
bgp router-id 10.95.51.81
! redistribute ospf
! redistribute connected
! network 10.95.51.0/25
!
!PT
neighbor 192.168.16.6 remote-as 41
neighbor 192.168.16.6 ebgp-multihop 10
! neighbor 192.168.16.6 capability route-refresh

!Intra-bgp
neighbor 10.95.51.110 remote-as 51
! neighbor 10.95.51.85 ebgp-multihop 10
! neighbor 10.95.51.85 capability route-refresh
network 10.95.51.0/25
 !
! IPv6 Section
 address-family ipv6
! network 2A02:9008:0:1910::/60
! redistribute connected
! redistribute ospf6
neighbor 192.168.16.6 activate
neighbor 10.95.51.110 activate
 network 2A02:9008:0:1910::/60
exit-address-family

access-list all permit any
!
line vty
```

- ospfd.conf

```
hostname CAFE-TIDPT
password zebra
!
!
!interface dummy0
!
!interface eth0
! ipospfmtu-ignore
```

```
!
!interface eth1
! ipospfmtu-ignore
!
!interface lo
!
interface eth1
interface eth2
!interface eth3

router ospf
ospf router-id 10.95.51.65
 redistribute connected
 redistribute bgp
 network 10.95.51.0/25 area 0.0.0.1
!
line vty
```

- ospf6d.conf

```
hostname ospf6d@CAFE-TIDpt
password zebra
log stdout
service advanced-vty
!
!debug ospf6 neighbor state
!
interface eth1
 ipv6 ospf6 cost 1
!
interface eth2
 ipv6 ospf6 cost 1
!
interface eth3
 ipv6 ospf6 cost 1
!
!interface eth4
! ipv6 ospf6 cost 1
!
!
router ospf6
 router-id 10.95.51.65
 redistribute connected
 redistribute  bgp
 interface eth1 area 0.0.0.1
 interface eth2 area 0.0.0.1
 interface eth3 area 0.0.0.1
! interface eth4 area 0.0.0.1
line vty
 exec-timeout 0 0
```

A.1.10.2 CAFE-TIDWUTConfiguration

- daemons

```
zebra=yes
bgpd=yes
```

```
ospfd=yes
ospf6d=yes
ripd=no
ripngd=no
isisd=no
```

- zebra

```
! -*- zebra -*-
!
! zebra sample configuration file
!
! $Id: zebra.conf.sample,v 1.1 2002/12/13 20:15:30 paulExp $
!
hostname router
password zebra
enable password zebra
!
! Interface's description.
interface br0
 link-detect
interface eth1
 ipv6 nd suppress-ra
 link-detect
interface eth2
 ipv6 nd suppress-ra
 link-detect
interface eth3
 ipv6 nd suppress-ra
 link-detect
!interface lo
! description test of desc.
!interface sit0
! multicast
! Static default route sample.
!
!ip route 0.0.0.0/0 203.181.89.241
!log file /var/log/quagga/zebra.log
ip forwarding
ipv6 forwarding
```

- bgpd.conf

```
! BGP configurtion for COMET federated testbed
! border router in TID
! 2012/02/08 18:43:00
!
hostname TID-WUT2
password zebra
!
log file /var/log/quagga/bgpd.log
!
!debug bgp updates
!
router bgp 51
bgp router-id 10.95.51.85
! redistribute ospf
! redistribute connected
! network 10.95.51.0/25
!
!WUT2
neighbor 10.95.51.109 remote-as 51
neighbor 10.203.62.249 remote-as 62
! neighbor 10.203.62.249 capability route-refresh
neighbor 10.203.62.249 ebgp-multihop 10
 !
network 10.95.51.0/25

! Intra-bgp
! neighbor 10.95.51.109 remote-as 51
! neighbor 10.95.51.109 capability route-refresh
! neighbor 10.95.51.109 ebgp-multihop 10
!

! IPv6 Section
 address-family ipv6
! network 2A02:9008:0:1910::/60
! redistribute connected
neighbor 10.95.51.109 activate
neighbor 10.203.62.249 activate
 network 2A02:9008:0:1910::/60
 exit-address-family

access-list all permit any
!
line vty
!
```

- ospfd.conf

```
! Zebra configuration saved from vty
!   2012/02/11 20:40:38
!
hostname CAFEedge2
password zebra
!
!
!interface dummy0
```

```
!
!interface eth0
! ipospfmtu-ignore
!
!interface eth1
! ipospfmtu-ignore
!
!interface lo
!
interface eth1
interface eth2
interface eth3

router ospf
ospf router-id 10.95.51.85
 redistribute connected
 redistribute bgp
 network 10.95.51.0/25 area 0.0.0.1
!
line vty
```

- ospf6d.conf

```
!
! Zebra configuration saved from vty
!   2003/11/28 00:49:49
!
hostname ospf6d@CAFE-TIDwut
password zebra
log stdout
service advanced-vty
!
!debug ospf6 neighbor state
!
interface eth1
 ipv6 ospf6 cost 1
!
interface eth2
 ipv6 ospf6 cost 1
!
interface eth3
 ipv6 ospf6 cost 1
! ipv6 nd suppress-ra
! link-detect
!
!interface eth4
! ipv6 ospf6 cost 1
!
!
router ospf6
 router-id 10.95.51.85
 redistribute connected
 redistribute bgp
 interface eth1 area 0.0.0.1
 interface eth2 area 0.0.0.1
 interface eth3 area 0.0.0.1
```

```
! interface eth4 area 0.0.0.1
!
!
line vty
 exec-timeout 0 0
```

A.1.10.3 CAFE-TID3Configuration

- daemons

```
zebra=yes
bgpd=no
ospfd=yes
ospf6d=yes
ripd=no
ripngd=no
isisd=no
```

- zebra.conf

```
! -*- zebra -*-
!
! zebra sample configuration file
!
! $Id: zebra.conf.sample,v 1.1 2002/12/13 20:15:30 paulExp $
!
hostname Router
password zebra
enable password zebra
!
! Interface's description.
!
!interface lo
! description test of desc.
!
!interface sit0
! multicast

interface eth1
 ipv6 nd suppress-ra
 link-detect
interface eth2
 ipv6 nd suppress-ra
 link-detect
!
! Static default route sample.
!
!ip route 0.0.0.0/0 203.181.89.241
!log file /var/log/quagga/zebra.log
ip forwarding
ipv6 forwarding
```

- ospfd.conf

```
hostname CAFE-TID3
password zebra
!interface dummy0

!interface eth0
! ipospfmtu-ignore
!
!interface eth1
! ipospfmtu-ignore
!
!interface lo
!
interface eth1
interface eth2
router ospf
ospf router-id 10.95.51.70
 redistribute connected
 redistribute bgp
 network 10.95.51.0/25 area 0.0.0.1
line vty
```

- ospf6d.conf!

```
hostname ospf6@CAFE-TID3
password zebra
log stdout
service advanced-vty
!
!debug ospf6 neighbor state
!
 interface eth1
 ipv6 ospf6 cost 1
!
 interface eth2
 ipv6 ospf6 cost 1
!
!interface eth3
! ipv6 ospf6 cost 1
!
!interface eth4
! ipv6 ospf6 cost 1
!
!
router ospf6
 router-id 10.95.51.70
 redistribute connected
! interface eth1 area 0.0.0.1
 interface eth2 area 0.0.0.1
! interface eth3 area 0.0.0.1
! interface eth4 area 0.0.0.1
!
line vty
 exec-timeout 0 0
```

A.1.10.4 CAFE-TID2Configuration

- daemons

```
zebra=yes
bgpd=no
ospfd=yes
ospf6d=yes
ripd=no
ripngd=no
isisd=no
```

- zebra.conf

```
hostname Router
password zebra
enable password zebra
!
! Interface's description.
!
!interface lo
! description test of desc.
interface br1
ip address 10.95.51.41/29
    link-detect
    no ipv6 nd suppress-ra
    ipv6 address 2A02:9008:0:1912:0:50:56A3:54/64
    ipv6 nd prefix 2A02:9008:0:1912::/64
interface eth3
ipv6 nd suppress-ra
link-detect
!
!interface sit0
! multicast
!
! Static default route sample.
!
!ip route 0.0.0.0/0 203.181.89.241
!
ip forwarding
ipv6 forwarding
!log file /var/log/quagga/zebra.log
```

- ospfd.conf

```
!
! Zebra configuration saved from vty
!   2012/02/11 20:40:38
!
hostname CAFE-TID2
password zebra
!
!
!interface dummy0
!
!interface eth0
! ipospfmtu-ignore
!
!interface eth1
! ipospfmtu-ignore
!
!interface lo
!
interface eth1
interface eth2
interface eth3

router ospf
ospf router-id 10.95.51.33
 redistribute connected
 redistribute bgp
 network 10.95.51.0/25 area 0.0.0.1
!
line vty
```

- ospf6d.conf

```
! Zebra configuration saved from vty
!   2003/11/28 00:49:49
!
hostname ospf6d@CAFE-TID2
password zebra
log stdout
service advanced-vty
!
!debug ospf6 neighbor state
!
interface eth1
 ipv6 ospf6 cost 1
!
interface eth2
 ipv6 ospf6 cost 1
!
interface eth3
 ipv6 ospf6 cost 1
!
!interface eth4
! ipv6 ospf6 cost 1
!
!
```

```
router ospf6
 router-id 10.95.51.33
 redistribute connected
 interface eth1 area 0.0.0.1
 interface eth2 area 0.0.0.1
 interface eth3 area 0.0.0.1
! interface eth4 area 0.0.0.1
!
!
line vty
 exec-timeout 0 0
```

A.2 WUT LOCAL TESTBED

A.2.1 Addressing Schema

Machine (AS number)	Interface	MAC Address	IPv4	IPv6 (global)
CAFE_A (AS 62)	eht0 eth1 eth2 eth3 lo	08:00:27:D3:D0:EF 08:00:27:B4:4C:F4 08:00:27:6A:91:2B 08:00:27:37:FE:15 X	192.168.56.21/24 10.203.62.227/28 10.203.62.249/29 10.203.62.246/31 X	X 2001:67c:24cc:31d1::cafe:a/64 X X 2001:67c:24cc:31d0::cafe:a/128
CAFE_B (AS 62)	eht0 eth1 eth2 lo	08:00:27:67:18:A4 08:00:27:5D:2D:1C 08:00:27:16:4F:09 X	192.168.56.22/24 10.203.62.225/28 10.203.62.1/27 X	X X 2001:67c:24cc:31d2::cafe:b/64 2001:67c:24cc:31d0::cafe:b/128
CAFE_C (AS 62)	eht0 eth1 eth2 lo	08:00:27:29:9A:68 08:00:27:95:31:CA 08:00:27:31:7D:18 X	192.168.56.23/24 10.203.62.226/28 10.203.61.247/31 X	X X X 2001:67c:24cc:31d0::cafe:c/128
CAFE_62 (AS 62)	eht0 eth1 eth2 lo	08:00:27:33:2B:70 08:00:27:DC:C6:C5 08:00:27:9E:16:2D X	192.168.56.25/24 10.203.62.229/28 10.203.62.241/30 X	X X 2001:67c:24cc:31d3::cafe:62/64 2001:67c:24cc:31d0::cafe:e2/128
AE_2 (AS 62)	eht0 eth1	08:00:27:08:93:fb 08:00:27:10:fb:4a	192.168.56.24/24 10.203.62.228/30	X 2001:67c:24cc:31d1::ae:2/64
CC_2 (AS 62)	eth2	X (dynamic)	10.203.62.2/27	X (auto-configuration)
CS_2 (AS 62)	eht0 eth2	08:00:27:9a:12:5d 08:00:27:24:6b:f7	192.168.56.242/24 10.203.62.242/30	X 2001:67c:24cc:31d3::242/64
CAFE_D (AS 61)	eht0 eth1 eth2 eth3 lo	08:00:27:49:A0:B0 08:00:27:06:9C:2F 08:00:27:E0:C2:46 08:00:27:EE:27:A2 X	192.168.56.11/24 10.203.61.227/28 192.168.17.2/24 10.203.63.247/31 X	X 2001:67c:24cc:31c1::cafe:d/64 X X 2001:67c:24cc:31c0::cafe:d/128
CAFE_E (AS 61)	eht0 eth1 eth2 lo	08:00:27:3A:3C:21 08:00:27:87:EA:2F 08:00:27:6B:B6:FF X	192.168.56.12/24 10.203.61.225/28 10.203.61.1/27 X	X X 2001:67c:24cc:31c2::cafe:e/64 2001:67c:24cc:31c0::cafe:e/128
CAFE_F (AS 61)	eht0 eth1 eth2 lo	08:00:27:B8:FF:AE 08:00:27:C6:4F:CB 08:00:27:8B:70:9F X	192.168.56.13/24 10.203.61.226/28 10.203.61.246/31 X	X X X 2001:67c:24cc:31c0::cafe:f/128
CAFE_61 (AS 61)	eht0 eth1 eth2 lo	08:00:27:DD:23:E5 08:00:27:B5:B4:2D 08:00:27:1E:7E:95 X	192.168.56.15/24 10.203.61.229/28 10.203.61.241/30 X	X X 2001:67c:24cc:31c3::cafe:62/64 2001:67c:24cc:31c0::cafe:e2/128
AE_1 (AS 62)	eht0 eth1	08:00:27:c5:fa:ba 08:00:27:7d:3c:99	192.168.56.14/24 10.203.61.228/30	X 2001:67c:24cc:31c1::ae:2/64
CC_1 (AS 62)	eth2	X (dynamic)	10.203.61.2/27	X (auto-configuration)

Machine (AS number)	Interface	MAC Address	IPv4	IPv6 (global)
CS_1 (AS 62)	eht0 eth2	08:00:27:8b:33:dc 08:00:27:16:4a:1f	192.168.56.243/24 10.203.61.242/30	X 2001:67c:24cc:31c3::242/64
CAFE_G (AS 63)	eht0 eth1 eth2 eth3 lo	08:00:27:C5:F2:8D 08:00:27:4B:61:28 08:00:27:D3:2E:B5 08:00:27:59:9B:F4 X	192.168.56.31/24 10.203.63.225/28 10.203.63.246/31 10.203.62.247/31 X	X X 2001:67c:24cc:31e1::cafe:9/64 X 2001:67c:24cc:31e0::cafe:9/128
AE_3	eht0 eth1	08:00:27:11:a8:d4 08:00:27:41:dc:e2	192.168.56.32/24 10.203.63.226/28	X 2001:67c:24cc:31e1::ae:3/64

Table 47: WUT Addressing Schema

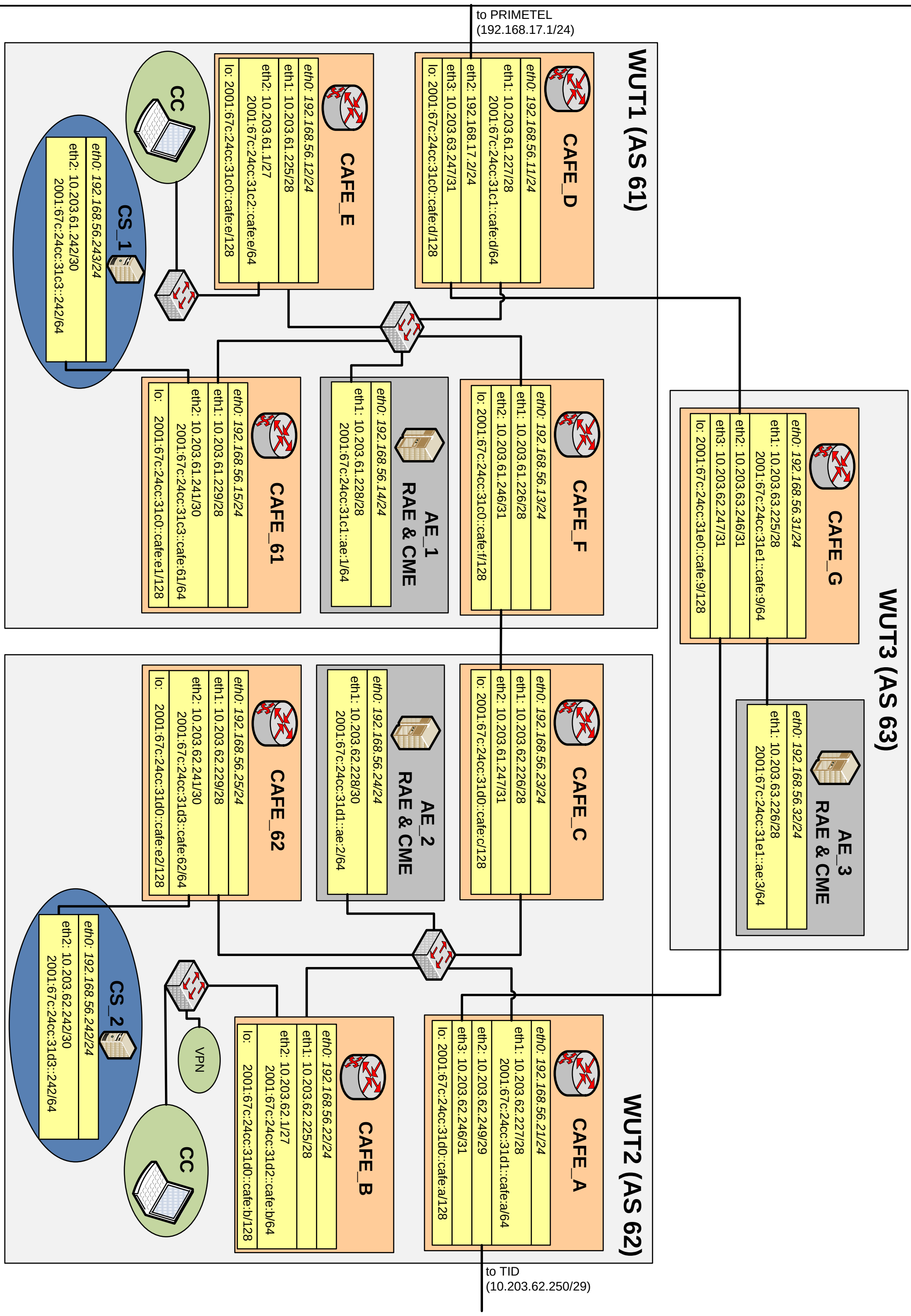


Figure 119: Detailed network diagram ofWUT Local Testbed

A.2.2 Forwarding Keys

CAFE	Key Name	Interface (device)	Destination MAC Address	Remote cafe associated	Remote interface
CAFE_A	0xeb	Eth1	08:00:27:5D:2D:1C	CAFE_B	Eth1
CAFE_A	0xec	Eth1	08:00:27:95:31:CA	CAFE_C	Eth1
CAFE_A	0xe6	Eth1	08:00:27:DC:C6:C5	CAFE_62	Eth1
CAFE_A	0x63	Eth3	08:00:27:59:9B:F4	CAFE_G	Eth3
CAFE_A	0x51	Eth2	DA:08:9A:71:9D:9F	CAFE_TIDWUT	Eth
CAFE_C	0xea	Eth1	08:00:27:B4:4C:F4	CAFE_A	Eth1
CAFE_C	0xeb	Eth1	08:00:27:5D:2D:1C	CAFE_B	Eth1
CAFE_C	0xe6	Eth1	08:00:27:DC:C6:C5	CAFE_62	Eth1
CAFE_C	0x61	Eth2	08:00:27:8B:70:9F	CAFE_F	Eth2
CAFE_62	0xea	Eth1	08:00:27:B4:4C:F4	CAFE_A	Eth1
CAFE_62	0xeb	Eth1	08:00:27:5D:2D:1C	CAFE_B	Eth1
CAFE_62	0xec	Eth1	08:00:27:95:31:CA	CAFE_C	Eth1
CAFE_D	0xef	Eth1	08:00:27:C6:4F:CB	CAFE_F	Eth1
CAFE_D	0xee	Eth1	08:00:27:87:EA:2F	CAFE_E	Eth1
CAFE_D	0xe6	Eth1	08:00:27:B5:B4:2D	CAFE_61	Eth1
CAFE_D	0x63	Eth3	08:00:27:D3:2E:B5	CAFE_G	Eth2
CAFE_D	0x41	Eth2	3A:45:A2:CD:5E:46	CAFE_PTWUT	Eth3
CAFE_F	0xed	Eth1	08:00:27:06:9C:2F	CAFE_D	Eth1
CAFE_F	0xee	Eth1	08:00:27:87:EA:2F	CAFE_E	Eth1
CAFE_F	0x61	Eth1	08:00:27:B5:B4:2D	CAFE_61	Eth1
CAFE_F	0x62	Eth2	08:00:27:31:7D:18	CAFE_C	Eth2

CAFE	Key Name	Interface (device)	Destination MAC Address	Remote cafe associated	Remote interface
CAFE_61	0xed	Eth1	08:00:27:06:9C:2F	CAFE_D	Eth1
CAFE_61	0xee	Eth1	08:00:27:87:EA:2F	CAFE_E	Eth1
CAFE_61	0xef	Eth1	08:00:27:C6:4F:CB	CAFE_F	Eth1
CAFE_G	0x62	Eth3	08:00:27:37:FE:15	CAFE_A	Eth3
CAFE_G	0x61	Eth2	08:00:27:EE:27:A2	CAFE_D	Eth3

Table 48: WUT Forwarding Keys

A.2.3 IP Prefixes/CAFE Mapping

Prefix	Prefix Length	Edge Cafe
10.203.61.0	27	CAFE_E
10.203.62.0	27	CAFE_B

Table 49: WUT Subnet/CAFE Mapping in IPv4

Prefix	Prefix Length	Edge Cafe
2001:67c:24cc:31c2::	64	CAFE_E
2001:67c:24cc:31d2::	64	CAFE_B

Table 50: WUT Subnet/CAFE Mapping in IPv6

A.2.4 Path Configuration in RAEs

AS/Range Source	AS/Range sink	COS	BW [MB]	IPLR	IPTD [s]
**************************AS 61**************************					
62	41	BtBE	100	0.000001	0.001
62	41	PR	100	0.000001	0.0005
41	62	BtBE	100	0.000001	0.001
41	62	PR	100	0.000001	0.0005
62	63	BtBE	100	0.000001	0.001
62	63	PR	100	0.000001	0.0005
63	62	BtBE	100	0.000001	0.001
63	62	PR	100	0.000001	0.0005
41	63	BtBE	100	0.000001	0.001
41	63	PR	100	0.000001	0.0005
63	41	BtBE	100	0.000001	0.001
63	41	PR	100	0.000001	0.0005
10.203.61.0/27	41	BtBE	100	0.000001	0.001
10.203.61.0/27	41	PR	100	0.000001	0.0005
41	10.203.61.0/27	BtBE	100	0.000001	0.001
41	10.203.61.0/27	PR	100	0.000001	0.0005
10.203.61.0/27	62	BtBE	100	0.000001	0.001
10.203.61.0/27	62	PR	100	0.000001	0.0005
62	10.203.61.0/27	BtBE	100	0.000001	0.001
62	10.203.61.0/27	PR	100	0.000001	0.0005

AS/Range Source	AS/Range sink	COS	BW [MB]	IPLR	IPTD [s]
10.203.61.0/27	63	BtBE	100	0.000001	0.001
10.203.61.0/27	63	PR	100	0.000001	0.0005
63	10.203.61.0/27	BtBE	100	0.000001	0.001
63	10.203.61.0/27	PR	100	0.000001	0.0005
10.203.61.0/27	10.203.61.240/30	BtBE	1000	0.0000001	0.0005
10.203.61.0/27	10.203.61.240/30	PR	1000	0.0000001	0.0005
10.203.61.240/30	41	BtBE	100	0.000001	0.001
10.203.61.240/30	41	PR	100	0.000001	0.0005
41	10.203.61.240/30	BtBE	100	0.000001	0.001
41	10.203.61.240/30	PR	100	0.000001	0.0005
10.203.61.240/30	62	BtBE	100	0.000001	0.001
10.203.61.240/30	62	PR	100	0.000001	0.0005
62	10.203.61.240/30	BtBE	100	0.000001	0.001
62	10.203.61.240/30	PR	100	0.000001	0.0005
10.203.61.240/30	63	BtBE	100	0.000001	0.001
10.203.61.240/30	63	PR	100	0.000001	0.0005
63	10.203.61.240/30	BtBE	100	0.000001	0.001
63	10.203.61.240/30	PR	100	0.000001	0.0005
10.203.61.240/30	10.203.61.0/27	BtBE	1000	0.0000001	0.0005
10.203.61.240/30	10.203.61.0/27	PR	1000	0.0000001	0.0005
**********************AS 62*************************					
61	51	BtBE	100	0.000001	0.001

AS/Range Source	AS/Range sink	COS	BW [MB]	IPLR	IPTD [s]
61	51	PR	100	0.000001	0.0005
51	61	BtBE	100	0.000001	0.001
51	61	PR	100	0.000001	0.0005
61	63	BtBE	100	0.000001	0.001
61	63	PR	100	0.000001	0.0005
63	61	BtBE	100	0.000001	0.001
63	61	PR	100	0.000001	0.0005
51	63	BtBE	100	0.000001	0.001
51	63	PR	100	0.000001	0.0005
63	51	BtBE	100	0.000001	0.001
63	51	PR	100	0.000001	0.0005
10.203.62.0/27	51	BtBE	100	0.000001	0.001
10.203.62.0/27	51	PR	100	0.000001	0.0005
51	10.203.62.0/27	BtBE	100	0.000001	0.001
51	10.203.62.0/27	PR	100	0.000001	0.0005
10.203.62.0/27	61	BtBE	100	0.000001	0.001
10.203.62.0/27	61	PR	100	0.000001	0.0005
61	10.203.62.0/27	BtBE	100	0.000001	0.001
61	10.203.62.0/27	PR	100	0.000001	0.0005
10.203.62.0/27	63	BtBE	100	0.000001	0.001
10.203.62.0/27	63	PR	100	0.000001	0.0005
63	10.203.62.0/27	BtBE	100	0.000001	0.001

AS/Range Source	AS/Range sink	COS	BW [MB]	IPLR	IPTD [s]
63	10.203.62.0/27	PR	100	0.000001	0.0005
10.203.62.0/27	10.203.62.240/30	BtBE	1000	0.0000001	0.0005
10.203.62.0/27	10.203.62.240/30	PR	1000	0.0000001	0.0005
10.203.62.240/30	51	BtBE	100	0.000001	0.001
10.203.62.240/30	51	PR	100	0.000001	0.0005
51	10.203.62.240/30	BtBE	100	0.000001	0.001
51	10.203.62.240/30	PR	100	0.000001	0.0005
10.203.62.240/30	61	BtBE	100	0.000001	0.001
10.203.62.240/30	61	PR	100	0.000001	0.0005
61	10.203.62.240/30	BtBE	100	0.000001	0.001
61	10.203.62.240/30	PR	100	0.000001	0.0005
10.203.62.240/30	63	BtBE	100	0.000001	0.001
10.203.62.240/30	63	PR	100	0.000001	0.0005
63	10.203.62.240/30	BtBE	100	0.000001	0.001
63	10.203.62.240/30	PR	100	0.000001	0.0005
10.203.62.240/30	10.203.62.0/27	BtBE	1000	0.0000001	0.0005
10.203.62.240/30	10.203.62.0/27	PR	1000	0.0000001	0.0005
****************************AS 63******************************					
62	61	BtBE	100	0.000001	0.001
62	61	PR	100	0.000001	0.0005
61	62	BtBE	100	0.000001	0.001

AS/Range Source	AS/Range sink	COS	BW [MB]	IPLR	IPTD [s]
61	62	PR	100	0.000001	0.0005

Table 51: WUT Path Configuration in RAEs

A.2.5 Peering AS/CAFE Mappings

AS Path	CoS	Source CAFE	Sink CAFE	Peer CME	BW [MB]
*****************AS 61*************************					
61,62	PR	10.203.61.226	10.203.62.226	10.203.62.228	100
61,62	BTBE	10.203.61.226	10.203.62.226	10.203.62.228	100
61,41	PR	10.203.61.227	10.50.50.2	10.50.50.6	100
61,41	BTBE	10.203.61.227	10.50.50.2	10.50.50.6	100
61,63	PR	10.203.61.227	10.203.63.225	10.203.63.226	100
61,63	BTBE	10.203.61.227	10.203.63.225	10.203.63.226	100
**************************AS 62******************************					
62,61	PR	10.203.62.226	10.203.61.226	10.203.61.228	100
62,61	BTBE	10.203.62.226	10.203.61.226	10.203.61.228	100
62,51	PR	10.203.62.227	10.95.51.85	10.95.51.3	100
62,51	BTBE	10.203.62.227	10.95.51.85	10.95.51.3	100
62,63	PR	10.203.62.227	10.203.63.225	10.203.63.226	100
62,63	BTBE	10.203.62.227	10.203.63.225	10.203.63.226	100
****************************AS 63****************************					
63,61	PR	10.203.63.225	10.203.61.227	10.203.61.228	100
63,61	BTBE	10.203.63.225	10.203.61.227	10.203.61.228	100
63,62	PR	10.203.63.225	10.203.62.227	10.203.62.228	100
63,62	BTBE	10.203.63.225	10.203.62.227	10.203.62.228	100

Table 52: WUT Peering AS/CAFE Mappings for IPv4

AS Path	CoS	Source CAFE	Sink CAFE	Peer CME	BW [MB]
*************************AS 61****************************					
61,62	PR	2001:67c:24cc:31c0::cafe:f	2001:67c:24cc:31d0::cafe:c	2001:67c:24cc:31d1::ae:2	100
61,62	BTBE	2001:67c:24cc:31c0::cafe:f	2001:67c:24cc:31d0::cafe:c	2001:67c:24cc:31d1::ae:2	100
61,41	PR	2001:67c:24cc:31c0::cafe:d	3105::2	3105::6	100
61,41	BTBE	2001:67c:24cc:31c0::cafe:d	3105::2	3105::6	100
61,63	PR	2001:67c:24cc:31c0::cafe:d	2001:67c:24cc:31e1::cafe:9	2001:67c:24cc:31e1::ae:3	100
61,63	BTBE	2001:67c:24cc:31c0::cafe:d	2001:67c:24cc:31e1::cafe:9	2001:67c:24cc:31e1::ae:3	100
**********************AS 62************************					
62,61	PR	2001:67c:24cc:31d0::cafe:c	2001:67c:24cc:31c0::cafe:f	2001:67c:24cc:31c1::ae:1	100
62,61	BTBE	2001:67c:24cc:31d0::cafe:c	2001:67c:24cc:31c0::cafe:f	2001:67c:24cc:31c1::ae:1	100
62,51	PR	2001:67c:24cc:31d0::cafe:a	2a02:9008:0:191:0:50:56a3:5f	2a02:9008:0:191:0:50:56a3:4f	100
62,51	BTBE	2001:67c:24cc:31d0::cafe:a	2a02:9008:0:191:0:50:56a3:5f	2a02:9008:0:191:0:50:56a3:4f	100
62,63	PR	2001:67c:24cc:31d0::cafe:a	2001:67c:24cc:31e0::cafe:9	2001:67c:24cc:31e1::ae:3	100
62,63	BTBE	2001:67c:24cc:31d0::cafe:a	2001:67c:24cc:31e0::cafe:9	2001:67c:24cc:31e1::ae:3	100
*************************AS 63***************************					
63,61	PR	2001:67c:24cc:31e0::cafe:9	2001:67c:24cc:31c0::cafe:d	10.203.61.228	100
63,61	BTBE	2001:67c:24cc:31e0::cafe:9	2001:67c:24cc:31c0::cafe:d	10.203.61.228	100
63,62	PR	2001:67c:24cc:31e0::cafe:9	2001:67c:24cc:31d0::cafe:a	2001:67c:24cc:31d1::ae:2	100
63,62	BTBE	2001:67c:24cc:31e0::cafe:9	2001:67c:24cc:31d0::cafe:a	2001:67c:24cc:31d1::ae:2	100

AS Path	CoS	Source CAFE	Sink CAFE	Peer CME	BW [MB]
****************************AS 61**********************************					

AS Path	CoS	Source CAFE	Sink CAFE	Peer CME	BW [MB]
61,62	PR	10.203.61.226	10.203.62.226	10.203.62.228	100000000
61,62	BTBE	10.203.61.226	10.203.62.226	10.203.62.228	100000000
61,41	PR	10.203.61.227	10.50.50.2	10.50.50.6	100000000
61,41	BTBE	10.203.61.227	10.50.50.2	10.50.50.6	100000000
61,63	PR	10.203.61.227	10.203.63.225	10.203.63.226	100000000
61,63	BTBE	10.203.61.227	10.203.63.225	10.203.63.226	100000000
************************AS 62******************************					
62,61	PR	10.203.62.226	10.203.61.226	10.203.61.228	100000000
62,61	BTBE	10.203.62.226	10.203.61.226	10.203.61.228	100000000
62,51	PR	10.203.62.227	10.95.51.85	10.95.51.3	100000000
62,51	BTBE	10.203.62.227	10.95.51.85	10.95.51.3	100000000
62,63	PR	10.203.62.227	10.203.63.225	10.203.63.226	100000000
62,63	BTBE	10.203.62.227	10.203.63.225	10.203.63.226	100000000
**********************AS 62**********************					
63,61	PR	10.203.63.225	10.203.61.227	10.203.61.228	100000000
63,61	BTBE	10.203.63.225	10.203.61.227	10.203.61.228	100000000
63,62	PR	10.203.63.225	10.203.62.227	10.203.62.228	100000000
63,62	BTBE	10.203.63.225	10.203.62.227	10.203.62.228	100000000

Table 53: WUT Peering AS/CAFE Mappings for IPv6

A.2.6 Peering CAFEs/Keys Mappings

Type	$CAFE_{or}$→$CAFE_{dest}$	Key list (reverse order)	CoS	BW
***********************AS 61**************************				
TRANSIT	10.203.61.241→ 10.203.61.1	0xee	Pr	1000M
TRANSIT	10.203.61.241→ 10.203.61.1	0xbe	BtBE	1000M
TRANSIT	10.203.61.241→ 10.203.61.227	0xed	Pr	100M
TRANSIT	10.203.61.241→ 10.203.61.227	0xbe	BtBE	100M
TRANSIT	10.203.61.241→ 10.203.61.226	0xef	Pr	100M
TRANSIT	10.203.61.241→ 10.203.61.226	0xbf	BtBE	100M
TRANSIT	10.203.61.241→ 10.203.61.227	0xed	Pr	100M
TRANSIT	10.203.61.241→ 10.203.61.227	0xbd	BtBE	100M
TRANSIT	10.203.61.227→10.203.61.241	0xe6	Pr	100M
TRANSIT	10.203.61.227→10.203.61.241	0xb6	BtBE	100M
TRANSIT	10.203.61.227→10.203.61.1	0xee	Pr	100M
TRANSIT	10.203.61.227→10.203.61.1	0xbe	BtBE	100M
TRANSIT	10.203.61.226→10.203.61.241	0x61	Pr	100M
TRANSIT	10.203.61.226→10.203.61.241	0x91	BtBE	100M
TRANSIT	10.203.61.226→10.203.61.1	0xee	Pr	100M
TRANSIT	10.203.61.226→10.203.61.1	0xbe	BtBE	100M
TRANSIT	10.203.61.227→10.203.61.241	0xe6	Pr	100M
TRANSIT	10.203.61.227→10.203.61.241	0xb6	BtBE	100M
TRANSIT	10.203.61.227→10.203.61.1	0xee	Pr	100M
TRANSIT	10.203.61.227→10.203.61.1	0xbe	BtBE	100M

Type	$CAFE_{or} \rightarrow CAFE_{dest}$	Key list (reverse order)	CoS	BW
TRANSIT	10.203.61.227→10.203.61.226	0xef	Pr	100M
TRANSIT	10.203.61.227→10.203.61.226	0xbf	BtBE	100M
TRANSIT	10.203.61.226→10.203.61.227	0xed	Pr	100M
TRANSIT	10.203.61.226→10.203.61.227	0xbd	BtBE	100M
TRANSIT	10.203.61.227→10.203.61.227	null	Pr	100M
TRANSIT	10.203.61.227→10.203.61.227	null	BtBE	100M
PEERING	10.203.61.227 →10.50.50.2	0x41	PR	100MB
PEERING	10.203.61.227 →10.50.50.2	0x42	BtBE	100MB
PEERING	10.203.61.227 →10.203.63.225	0x63	PR	100MB
PEERING	10.203.61.227 →10.203.63.225	0x93	BtBE	100MB
PEERING	10.203.61.226 →10.203.62.226	0x62	PR	100MB
PEERING	10.203.61.226 →10.203.62.226	0x92	BtBE	100MB
*************************AS 62*************************				
TRANSIT	10.203.62.241→ 10.203.62.1	0xeb	Pr	1000M
TRANSIT	10.203.62.241→ 10.203.62.1	0xbb	BtBE	1000M
TRANSIT	10.203.62.241→ 10.203.62.227	0xea	Pr	100M
TRANSIT	10.203.62.241→ 10.203.62.227	0xba	BtBE	100M
TRANSIT	10.203.62.241→ 10.203.62.226	0xec	Pr	100M
TRANSIT	10.203.62.241→ 10.203.62.226	0xbc	BtBE	100M
TRANSIT	10.203.62.241→ 10.203.62.227	0xea	Pr	100M
TRANSIT	10.203.62.241→ 10.203.62.227	0xba	BtBE	100M
TRANSIT	10.203.62.227→10.203.62.241	0xe6	Pr	100M

Type	$CAFE_{or}$→$CAFE_{dest}$	Key list (reverse order)	CoS	BW
TRANSIT	10.203.62.227→10.203.62.241	0xb6	BtBE	100M
TRANSIT	10.203.62.227→10.203.62.1	0xeb	Pr	100M
TRANSIT	10.203.62.227→10.203.62.1	0xbb	BtBE	100M
TRANSIT	10.203.62.226→10.203.62.241	0x61	Pr	100M
TRANSIT	10.203.62.226→10.203.62.241	0x91	BtBE	100M
TRANSIT	10.203.62.226→10.203.62.1	0xeb	Pr	100M
TRANSIT	10.203.62.226→10.203.62.1	0xbb	BtBE	100M
TRANSIT	10.203.62.227→10.203.62.241	0xe6	Pr	100M
TRANSIT	10.203.62.227→10.203.62.241	0xb6	BtBE	100M
TRANSIT	10.203.62.227→10.203.62.1	0xeb	Pr	100M
TRANSIT	10.203.62.227→10.203.62.1	0xbb	BtBE	100M
TRANSIT	10.203.62.227→10.203.62.226	0xec	Pr	100M
TRANSIT	10.203.62.227→10.203.62.226	0xbc	BtBE	100M
TRANSIT	10.203.61.226→10.203.61.227	0xea	Pr	100M
TRANSIT	10.203.61.226→10.203.61.227	0xba	BtBE	100M
TRANSIT	10.203.62.227→10.203.62.227	null	Pr	100M
TRANSIT	10.203.62.227→10.203.62.227	null	BtBE	100M
PEERING	10.203.62.227 →10.95.51.85	0x51	PR	100MB
PEERING	10.203.62.227 →10.95.51.85	0x52	BtBE	100MB
PEERING	10.203.62.227 →10.203.63.225	0x63	PR	100MB
PEERING	10.203.62.227 →10.203.63.225	0x93	BtBE	100MB
PEERING	10.203.62.226 →10.203.61.226	0x61	PR	100MB

Type	$CAFE_{or}$→$CAFE_{dest}$	Key list (reverse order)	CoS	BW
PEERING	10.203.62.226 →10.203.61.226	0x91	BtBE	100MB
***********************AS 63**********************				
TRANSIT	10.203.63.225→10.203.63.225	null	Pr	100M
TRANSIT	10.203.63.225→10.203.63.225	null	BtBE	100M
PEERING	10.203.62.225 →10.203.61.227	0x61	PR	100MB
PEERING	10.203.62.225 →10.203.61.227	0x91	BtBE	100MB
PEERING	10.203.62.225 →10.203.62.227	0x62	PR	100MB
PEERING	10.203.62.225 →10.203.62.227	0x92	BtBE	100MB

Table 54: WUT Peering CAFEs/Keys Mappings in IPv4

Type	$CAFE_{or}$→$CAFE_{dest}$	Key list (revers e order)	CoS	BW
AS 61				
TRANSIT	2001:67c:24cc:31c3::cafe:61→2001:67c:24cc:31c2::cafe:e	0xee	Pr	1000M
TRANSIT	2001:67c:24cc:31c3::cafe:61→2001:67c:24cc:31c2::cafe:e	0xbe	BtB E	1000M
TRANSIT	2001:67c:24cc:31c3::cafe:61→2001:67c:24cc:31c0::cafe:d	0xed	Pr	100M
TRANSIT	2001:67c:24cc:31c3::cafe:61→2001:67c:24cc:31c0::cafe:d	0xbe	BtB E	100M
TRANSIT	2001:67c:24cc:31c3::cafe:61→ 2001:67c:24cc:31c0::cafe:f	0xef	Pr	100M
TRANSIT	2001:67c:24cc:31c3::cafe:61→ 2001:67c:24cc:31c0::cafe:f	0xbf	BtB E	100M
TRANSIT	2001:67c:24cc:31c3::cafe:61→2001:67c:24cc:31c0::cafe:d	0xed	Pr	100M
TRANSIT	2001:67c:24cc:31c3::cafe:61→2001:67c:24cc:31c0::cafe:d	0xbd	BtB E	100M
TRANSIT	2001:67c:24cc:31c1::cafe:d→ 2001:67c:24cc:31c3::cafe:61	0xe6	Pr	100M
TRANSIT	2001:67c:24cc:31c1::cafe:d→ 2001:67c:24cc:31c3::cafe:61	0xb6	BtB E	100M
TRANSIT	2001:67c:24cc:31c1::cafe:d→2001:67c:24cc:31c2::cafe:e	0xee	Pr	100M

Type	$CAFE_{or}$→$CAFE_{dest}$	Key list (reverse order)	CoS	BW
TRANSIT	2001:67c:24cc:31c1::cafe:d→2001:67c:24cc:31c2::cafe:e	0xbe	BtBE	100M
TRANSIT	2001:67c:24cc:31c0::cafe:f→ 2001:67c:24cc:31c3::cafe:61	0x61	Pr	100M
TRANSIT	2001:67c:24cc:31c0::cafe:f→ 2001:67c:24cc:31c3::cafe:61	0x91	BtBE	100M
TRANSIT	2001:67c:24cc:31c0::cafe:f→2001:67c:24cc:31c2::cafe:e	0xee	Pr	100M
TRANSIT	2001:67c:24cc:31c0::cafe:f→2001:67c:24cc:31c2::cafe:e	0xbe	BtBE	100M
TRANSIT	2001:67c:24cc:31c1::cafe:d→ 2001:67c:24cc:31c3::cafe:61	0xe6	Pr	100M
TRANSIT	2001:67c:24cc:31c1::cafe:d→ 2001:67c:24cc:31c3::cafe:61	0xb6	BtBE	100M
TRANSIT	2001:67c:24cc:31c1::cafe:d→2001:67c:24cc:31c2::cafe:e	0xee	Pr	100M
TRANSIT	2001:67c:24cc:31c1::cafe:d→2001:67c:24cc:31c2::cafe:e	0xbe	BtBE	100M
TRANSIT	2001:67c:24cc:31c1::cafe:d→ 2001:67c:24cc:31c0::cafe:f	0xef	Pr	100M
TRANSIT	2001:67c:24cc:31c1::cafe:d→ 2001:67c:24cc:31c0::cafe:f	0xbf	BtBE	100M
TRANSIT	2001:67c:24cc:31c0::cafe:f→2001:67c:24cc:31c1::cafe:d	0xed	Pr	100M
TRANSIT	2001:67c:24cc:31c0::cafe:f→2001:67c:24cc:31c1::cafe:d	0xbd	BtBE	100M
TRANSIT	2001:67c:24cc:31c1::cafe:d→2001:67c:24cc:31c1::cafe:d	null	Pr	100M
TRANSIT	2001:67c:24cc:31c1::cafe:d→2001:67c:24cc:31c1::cafe:d	null	BtBE	100M
PEERING	2001:67c:24cc:31c1::cafe:d→3105::2	0x41	PR	100MB
PEERING	2001:67c:24cc:31c1::cafe:d→3105::2	0x42	BtBE	100MB
PEERING	2001:67c:24cc:31c1::cafe:d→2001:67c:24cc:31e0::cafe:9	0x63	PR	100MB
PEERING	2001:67c:24cc:31c1::cafe:d→2001:67c:24cc:31e0::cafe:9	0x93	BtBE	100MB
PEERING	2001:67c:24cc:31c0::cafe:f→2001:67c:24cc:31d0::cafe:c	0x62	PR	100MB
PEERING	2001:67c:24cc:31c0::cafe:f→2001:67c:24cc:31d0::cafe:c	0x92	BtBE	100MB

Type	$CAFE_{or}$→$CAFE_{dest}$	Key list (reverse order)	CoS	BW
AS 62				
TRANSIT	2001:67c:24cc:31d0::cafe:e2→2001:67c:24cc:31d0::cafe:b	0xeb	Pr	1000M
TRANSIT	2001:67c:24cc:31d0::cafe:e2→2001:67c:24cc:31d0::cafe:b	0xbb	BtBE	1000M
TRANSIT	2001:67c:24cc:31d0::cafe:e2→2001:67c:24cc:31d0::cafe:a	0xea	Pr	100M
TRANSIT	2001:67c:24cc:31d0::cafe:e2→2001:67c:24cc:31d0::cafe:a	0xba	BtBE	100M
TRANSIT	2001:67c:24cc:31d0::cafe:e2→2001:67c:24cc:31d0::cafe:c	0xec	Pr	100M
TRANSIT	2001:67c:24cc:31d0::cafe:e2→2001:67c:24cc:31d0::cafe:c	0xbc	BtBE	100M
TRANSIT	2001:67c:24cc:31d0::cafe:e2→2001:67c:24cc:31d0::cafe:a	0xea	Pr	100M
TRANSIT	2001:67c:24cc:31d0::cafe:e2→2001:67c:24cc:31d0::cafe:a	0xba	BtBE	100M
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:e2	0xe6	Pr	100M
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:e2	0xb6	BtBE	100M
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:b	0xeb	Pr	100M
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:b	0xbb	BtBE	100M
TRANSIT	2001:67c:24cc:31d0::cafe:c→ 2001:67c:24cc:31d0::cafe:e2	0x61	Pr	100M
TRANSIT	2001:67c:24cc:31d0::cafe:c→ 2001:67c:24cc:31d0::cafe:e2	0x91	BtBE	100M
TRANSIT	2001:67c:24cc:31d0::cafe:c→ 2001:67c:24cc:31d0::cafe:b	0xeb	Pr	100M
TRANSIT	2001:67c:24cc:31d0::cafe:c→ 2001:67c:24cc:31d0::cafe:b	0xbb	BtBE	100M
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:e2	0xe6	Pr	100M
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:e2	0xb6	BtBE	100M
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:b	0xeb	Pr	100M
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:b	0xbb	BtBE	100M
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:c	0xec	Pr	100M

Type	$CAFE_{or} \rightarrow CAFE_{dest}$	Key list (reverse order)	CoS	BW
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:c	0xbc	BtBE	100M
TRANSIT	2001:67c:24cc:31c0::cafe:f→2001:67c:24cc:31c1::cafe:d	0xea	Pr	100M
TRANSIT	2001:67c:24cc:31c0::cafe:f→2001:67c:24cc:31c1::cafe:d	0xba	BtBE	100M
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:a	null	Pr	100M
TRANSIT	2001:67c:24cc:31d0::cafe:a→ 2001:67c:24cc:31d0::cafe:a	null	BtBE	100M
PEERING	2001:67c:24cc:31d0::cafe:a→2a02:9008:0:1919:0:50:56a3:5f	0x51	PR	100MB
PEERING	2001:67c:24cc:31d0::cafe:a→2a02:9008:0:1919:0:50:56a3:5f	0x52	BtBE	100MB
PEERING	2001:67c:24cc:31d0::cafe:a→2001:67c:24cc:31e0::cafe:9	0x63	PR	100MB
PEERING	2001:67c:24cc:31d0::cafe:a→2001:67c:24cc:31e0::cafe:9	0x93	BtBE	100MB
PEERING	2001:67c:24cc:31d0::cafe:c→ 2001:67c:24cc:31c0::cafe:f	0x61	PR	100MB
PEERING	2001:67c:24cc:31d0::cafe:c→ 2001:67c:24cc:31c0::cafe:f	0x91	BtBE	100MB
AS 63				
TRANSIT	2001:67c:24cc:31e0::cafe:9→ 2001:67c:24cc:31e0::cafe:9	null	Pr	100M
TRANSIT	2001:67c:24cc:31e0::cafe:9→ 2001:67c:24cc:31e0::cafe:9	null	BtBE	100M
PEERING	2001:67c:24cc:31d0::cafe:b→2001:67c:24cc:31c1::cafe:d	0x61	PR	100MB
PEERING	2001:67c:24cc:31d0::cafe:b→2001:67c:24cc:31c1::cafe:d	0x91	BtBE	100MB
PEERING	2001:67c:24cc:31d0::cafe:b→2001:67c:24cc:31d0::cafe:a	0x62	PR	100MB
PEERING	2001:67c:24cc:31d0::cafe:b→2001:67c:24cc:31d0::cafe:a	0x92	BtBE	100MB

Table 55: WUT Peering CAFEs/Keys Mappings in IPv6

A.2.7 Content Client to COMET CoS Mappings

Host	Client IP	CoS
*****************AS 61*****************		
CC_1	10.203.61.2	PR
*****************AS 62*****************		
CC_2	10.203.62.2	PR

Table 56: WUT CC to CoS Mapping in IPv4

Host	Client IP	CoS
***************************AS 61****************************		
CC_1	2001:67c:24cc:31c2::2	PR
***************************AS 62****************************		
CC_2	2001:67c:24cc:31d2::2	PR

Table 57: WUT CC to CoS Mapping in IPv6

A.2.8 Content Names

Name	CoS	QoS	URL	MIME	CME
wut2.pl/cars_be	Be	BW:1Mbps Delay: 1 s Loss: 0.01	http://10.203.61.242:8888/get/127.ts http://10.203.62.242:8888/get/127.ts	video/mpeg	10.203.61.228 10.203.62.228
wut2.pl/cars_pr	Pr	BW:1Mbps Delay: 0.4 s Loss: 0.001	http://10.203.61.242:8899/get/127.ts http://10.203.62.242:8899/get/127.ts	video/mpeg	10.203.61.228 10.203.62.228
wut2.pl/sun_be	Be	BW:5Mbps Delay: 1 s Loss: 0.01	http://10.203.61.242:8888/get/103.ts http://10.203.62.242:8888/get/103.ts	video/mpeg	10.203.61.228 10.203.62.228
wut2.pl/sun_pr	Pr	BW:5Mbps Delay: 0.4 s Loss: 0.001	http://10.203.61.242:8899/get/103.ts http://10.203.62.242:8899/get/103.ts	video/mpeg	10.203.61.228 10.203.62.228
wut2.pl/match_be	Be	BW:131kbps Delay: 1 s Loss: 0.01	http://10.203.61.242:8769/stream.ts http://10.203.62.242:8769/stream.ts	video/mpeg	10.203.61.228 10.203.62.228
wut2.pl/match_pr	Pr	BW:131kbps Delay: 0.4 s Loss: 0.001	http://10.203.61.242:8769/stream.ts http://10.203.62.242:8769/stream.ts	video/mpeg	10.203.61.228 10.203.62.228

Table 58: WUT Content Names defined for IPv4

Name	CoS	QOS	URL	MIME	CME
wut2.pl/cars_be	Be	BW:1Mbps Delay: 1 s Loss: 0.01	http://10.203.61.242:8888/get/127.ts http://10.203.62.242:8888/get/127.ts	video/mpeg	2001:67c:24cc:31c1::ae:1 2001:67c:24cc:31d1::ae:2
wut2.pl/cars_pr	Pr	BW:1Mbps Delay: 0.4 s Loss: 0.001	http://10.203.61.242:8899/get/127.ts http://10.203.62.242:8899/get/127.ts	video/mpeg	2001:67c:24cc:31c1::ae:1 2001:67c:24cc:31d1::ae:2
wut2.pl/sun_be	Be	BW:5Mbps Delay: 1 s Loss: 0.01	http://10.203.61.242:8888/get/103.ts	video/mpeg	2001:67c:24cc:31c1::ae:1

			http://10.203.62.242:8888/get/103.ts		2001:67c:24cc:31d1::ae:2
wut2.pl/sun_pr	Pr	BW:5Mbps Delay: 0.4 s Loss: 0.001	http://10.203.61.242:8899/get/103.ts http://10.203.62.242:8899/get/103.ts	video/mpeg	2001:67c:24cc:31c1::ae:1 2001:67c:24cc:31d1::ae:2
wut2.pl/match_be	Be	BW:131kbps Delay: 1 s Loss: 0.01	http://10.203.61.242:8769/stream.ts http://10.203.62.242:8769/stream.ts	video/mpeg	2001:67c:24cc:31c1::ae:1 2001:67c:24cc:31d1::ae:2
wut2.pl/match_pr	Pr	BW:131kbps Delay: 0.4 s Loss: 0.001	http://10.203.61.242:8769/stream.ts http://10.203.62.242:8769/stream.ts	video/mpeg	2001:67c:24cc:31c1::ae:1 2001:67c:24cc:31d1::ae:2

Table 59: WUT Content Names defined for IPv6

A.2.9 OpenVPN Configuration

A.2.9.1 Configuration file of OpenVPN client at WUT side of PTL-WUT tunnel

```
# This VPN is used for COMET to connect to PRIMETELclient
dev tap0
proto udp
remote 217.27.59.123 2011
nobind
ca /etc/openvpn/keys-comet-ptl/ca.crt
cert /etc/openvpn/keys-comet-ptl/ptl-wut.crt
key /etc/openvpn/keys-comet-ptl/ptl-wut.key

#ns-cert-type server

# Use compression
comp-lzo
```

A.2.9.2 Configuration file of OpenVPN server at WUT side of WUT-TID tunnel

```
port 2001
proto udp
dev tap2
ca /etc/openvpn/keys-comet-tid/ca.crt
cert /etc/openvpn/keys-comet-tid/server.crt
key /etc/openvpn/keys-comet-tid/server.key
dh /etc/openvpn/keys-comet-tid/dh1024.pem

mode server
tls-server
#ifconfig 192.168.99.1 255.255.255.248
#ifconfig-pool 192.168.99.2 192.168.99.5 255.255.255.248
ifconfig-pool 10.203.62.26 10.203.62.30 255.255.255.224

push "route 10.203.60.0 255.255.252.0  10.203.62.1"
push "route 10.95.51.0 255.255.255.0 10.203.62.1"
push "route 10.50.50.0 255.255.255.0 10.203.62.1"
push "route 101.101.101.0 255.255.255.0 10.203.62.1"

keepalive 10 120
```

```
cipher none
```

A.2.10 Quagga Configuration

A.2.10.1 CAFE_D Configuration

- zebra.conf

```
! Zebra configuration saved from vty
!   2012/04/10 18:57:06
!
hostname CAFE_D
password zebra
enable password zebra
!
interface dummy0
 ipv6 nd suppress-ra
!
interface eth0
 ipv6 nd suppress-ra
!
interface eth1
 link-detect
 ipv6 nd suppress-ra
!
interface eth2
 link-detect
 ipv6 nd suppress-ra
!
interface eth3
 link-detect
 ipv6 nd suppress-ra
!
interface lo
 ipv6 address 2001:67c:24cc:31c0::cafe:d/128
!
ip forwarding
ipv6 forwarding
!
```

```
!
line vty
!
```

- bgpd.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:57:06
!
hostname CAFE_D
password zebra
!
router bgp 61
 bgp router-id 10.203.61.227
 network 10.203.61.0/24
 neighbor 10.203.61.226 remote-as 61
 neighbor 10.203.63.246 remote-as 63
 neighbor 10.203.63.246 ebgp-multihop 10
 # neighbor PRIMETEL
 neighbor 192.168.17.1 remote-as 41
 neighbor 192.168.17.1 ebgp-multihop 10
!
 address-family ipv6
 network 2001:67c:24cc:31c0::/60
 neighbor 10.203.61.226 activate
 neighbor 10.203.61.226 next-hop-self
 neighbor 10.203.63.246 activate
 neighbor 192.168.17.1  activate
 exit-address-family
!
line vty
!
```

- ospfd.conf

```
! Zebra configuration saved from vty
!   2012/04/10 18:57:06
!
hostname CAFE_D
```

```
password zebra
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface eth3
!
interface lo
!
router ospf
 ospf router-id 10.203.61.227
 redistribute connected
 redistribute bgp
 network 10.203.61.224/28 area 0.0.0.1
!
line vty
!
```

- ospf6d.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:57:06
!
hostname ospf6dCAFE_D
password zebra
!
debug ospf6 lsa unknown
!
interface eth1
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
```

```
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 10.203.61.227
 redistribute connected
 redistribute bgp
 interface eth1 area 0.0.0.1
!
line vty
!
```

A.2.10.2 CAFE_E Configuration

- zebra.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:52:24
!
hostname CAFE_E
password zebra
enable password zebra
!
interface dummy0
 ipv6 nd suppress-ra
!
interface eth0
 ipv6 nd suppress-ra
!
interface eth1
 link-detect
 ipv6 nd suppress-ra
!
interface eth2
 link-detect
 ipv6 nd suppress-ra
!
interface lo
```

```
 ipv6 address 2001:67c:24cc:31c0::cafe:e/128
!
ip forwarding
ipv6 forwarding

!
line vty
!
```

- ospfd.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:52:24
!
hostname CAFE_E
password zebra
!
!
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
router ospf
 ospf router-id 10.203.61.225
 redistribute connected
 network 10.203.61.224/28 area 0.0.0.1
!
line vty
!
```

- ospf6d.conf

```
!
```

```
! Zebra configuration saved from vty
!   2012/04/10 18:52:24
!
hostname ospf6d@CAFE_E
password zebra
!
debug ospf6 lsa unknown
!
interface eth1
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 10.203.61.225
 redistribute connected
 interface eth1 area 0.0.0.1
!
line vty
!
```

A.2.10.3 CAFE_F Configuration

- zebra.conf

```
!
! Zebra configuration saved from vty
!   2012/10/18 23:51:38
!
hostname CAFE_F
password zebra
enable password zebra
!
interface dummy0
```

```
 ipv6 nd suppress-ra
!
interface eth0
 ipv6 nd suppress-ra
!
interface eth1
 link-detect
 ipv6 nd suppress-ra
!
interface eth2
 link-detect
 ipv6 nd suppress-ra
!
interface lo
 ipv6 address 2001:67c:24cc:31c0::cafe:f/128
!
ip forwarding
ipv6 forwarding
!
line vty
!
```

- bgpd.conf

```
!
! Zebra configuration saved from vty
!   2012/10/18 23:51:38
!
hostname CAFE_F
password zebra
!
router bgp 61
 bgp router-id 10.203.61.226
 network 10.203.61.0/24
 neighbor 10.203.61.227 remote-as 61
 neighbor 10.203.61.247 remote-as 62
 neighbor 10.203.61.247 ebgp-multihop 10
!
```

```
 address-family ipv6
 network 2001:67c:24cc:31c0::/60
 neighbor 10.203.61.227 activate
 neighbor 10.203.61.227 next-hop-self
 neighbor 10.203.61.247 activate
 exit-address-family
!
line vty
!
```

- ospfd.conf

```
!
! Zebra configuration saved from vty
!   2012/10/18 23:51:38
!
hostname CAFE_F
password zebra
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
router ospf
 ospf router-id 10.203.61.226
 redistribute connected
 network 10.203.61.224/28 area 0.0.0.1
!
line vty
!
```

- ospf6d.conf

```
!
```

```
! Zebra configuration saved from vty
!   2012/10/18 23:51:38
!
hostname ospf6d@CAFE_F
password zebra
!
debug ospf6 lsa unknown
!
interface eth1
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 10.203.61.226
 redistribute connected
 interface eth1 area 0.0.0.1
!
line vty
!
```

A.2.10.4 CAFE_61 Configuration

- zebra.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:59:50
!
hostname CAFE_61
password zebra
enable password zebra
!
interface dummy0
 ipv6 nd suppress-ra
!
interface eth0
 ipv6 nd suppress-ra
!
interface eth1
 link-detect
 ipv6 nd suppress-ra
```

```
!
interface eth2
 link-detect
 ipv6 nd suppress-ra
!
interface lo
 ipv6 address 2001:67c:24cc:31c0::cafe:61/128
!
ip forwarding
ipv6 forwarding
!
!
line vty
!
```

- ospfd.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:59:50
!
hostname CAFE_61
password zebra
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
router ospf
 ospf router-id 10.203.61.229
 redistribute connected
 redistribute bgp
 network 10.203.61.224/28 area 0.0.0.1
!
line vty
!
```

- ospf6d.conf

```
!
! Zebra configuration saved from vty
```

```
!   2012/04/10 18:59:50
!
hostname ospf6d@CAFE_61
password zebra
!
debug ospf6 lsa unknown
!
interface eth1
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 10.203.61.229
 redistribute connected
 interface eth1 area 0.0.0.1
!
line vty
!
```

A.2.10.5 CAFE_A Configuration

- zebra.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 17:30:43
!
hostname CAFE_A
password zebra
enable password zebra
!
interface dummy0
 ipv6 nd suppress-ra
!
interface eth0
 ipv6 nd suppress-ra
!
interface eth1
 link-detect
 ipv6 nd suppress-ra
!
interface eth2
 link-detect
 ipv6 nd suppress-ra
!
interface eth3
 link-detect
 ipv6 nd suppress-ra
!
interface lo
 ipv6 address 2001:67c:24cc:31d0::cafe:a/128
!
```

```
ip forwarding
ipv6 forwarding
!
!
line vty
!
```

- bgpd.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 17:30:43
!
hostname CAFE_A
password zebra
!
router bgp 62
 bgp router-id 10.203.62.227
 network 10.203.62.0/24
 neighbor 10.203.62.226 remote-as 62
 neighbor 10.203.62.247 remote-as 63
 neighbor 10.203.62.247 ebgp-multihop 10
 neighbor 10.203.62.250 remote-as 51
 neighbor 10.203.62.250 ebgp-multihop 10
!
 address-family ipv6
 network 2001:67c:24cc:31d0::/60
 neighbor 10.203.62.226 activate
 neighbor 10.203.62.226 next-hop-self
 neighbor 10.203.62.247 activate
 neighbor 10.203.62.250 activate
 exit-address-family
!
line vty
!
```

- ospfd.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 17:30:43
!
hostname CAFE_A
password zebra
!
!
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface eth3
!
interface lo
!
```

```
router ospf
 ospf router-id 10.203.62.227
 redistribute connected
 redistribute bgp
 network 10.203.62.224/28 area 0.0.0.1
!
line vty
!
```

- ospf6d.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 17:30:43
!
hostname ospf6d@CAFE_A
password zebra
!
debug ospf6 lsa unknown
!
interface eth1
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 10.203.62.227
 redistribute connected
 redistribute bgp
 interface eth1 area 0.0.0.1
!
line vty
!
```

A.2.10.6 CAFE_B Configuration

- zebra.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:59:50
!
hostname CAFE_B
password zebra
enable password zebra
!
interface dummy0
 ipv6 nd suppress-ra
!
interface eth0
 ipv6 nd suppress-ra
!
interface eth1
 link-detect
 ipv6 nd suppress-ra
```

```
!
interface eth2
 link-detect
 ipv6 nd suppress-ra
!
interface lo
 ipv6 address 2001:67c:24cc:31d0::cafe:b/128
!
ip forwarding
ipv6 forwarding
!
!
line vty
!
```

- ospfd.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:59:50
!
hostname CAFE_B
password zebra
!
!
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
router ospf
 ospf router-id 10.203.62.225
 redistribute connected
 redistribute bgp
 network 10.203.62.224/28 area 0.0.0.1
!
line vty
!
```

- ospf6d.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:59:50
!
hostname ospf6d@CAFE_B
password zebra
!
debug ospf6 lsa unknown
!
interface eth1
 ipv6 ospf6 cost 1
```

```
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 10.203.62.225
 redistribute connected
 interface eth1 area 0.0.0.1
!
line vty
!
```

A.2.10.7 CAFE_C Configuration

- zebra.conf

```
!
! Zebra configuration saved from vty
!   2012/10/18 07:01:19
!
hostname CAFE_C
password zebra
enable password zebra
!
interface dummy0
 ipv6 nd suppress-ra
!
interface eth0
 ipv6 nd suppress-ra
!
interface eth1
 link-detect
 ipv6 nd suppress-ra
!
interface eth2
 link-detect
 ipv6 nd suppress-ra
!
interface lo
 ipv6 address 2001:67c:24cc:31d0::cafe:c/128
!
ip forwarding
ipv6 forwarding
!
!
line vty
!
```

- bgpd.conf

```
!
! Zebra configuration saved from vty
!   2012/10/18 07:01:19
!
hostname CAFE_C
password zebra
```

```
!
router bgp 62
 bgp router-id 10.203.62.226
 network 10.203.62.0/24
 neighbor 10.203.61.246 remote-as 61
 neighbor 10.203.61.246 ebgp-multihop 10
 neighbor 10.203.62.227 remote-as 62
!
 address-family ipv6
 network 2001:67c:24cc:31d0::/60
 neighbor 10.203.61.246 activate
 neighbor 10.203.62.227 activate
 neighbor 10.203.62.227 next-hop-self
 exit-address-family
!
line vty
!
```

- ospfd.conf

```
!
! Zebra configuration saved from vty
!   2012/10/18 07:01:19
!
hostname CAFE_C
password zebra
!
!
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
router ospf
 ospf router-id 10.203.62.226
 redistribute connected
 network 10.203.62.224/28 area 0.0.0.1
!
line vty
!
```

- ospf6d.conf

```
!
! Zebra configuration saved from vty
!   2012/10/18 07:01:19
!
hostname ospf6d@CAFE_C
password zebra
!
debug ospf6 lsa unknown
```

```
!
interface eth1
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 10.203.62.226
 redistribute connected
 interface eth1 area 0.0.0.1
!
line vty
!
```

A.2.10.8 CAFE_62 Configuration

- zebra.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:59:50
!
hostname CAFE_62
password zebra
enable password zebra
!
interface dummy0
 ipv6 nd suppress-ra
!
interface eth0
 ipv6 nd suppress-ra
!
interface eth1
 link-detect
 ipv6 nd suppress-ra
!
interface eth2
 link-detect
 ipv6 nd suppress-ra
!
interface lo
 ipv6 address 2001:67c:24cc:31d0::cafe:62/128
!
ip forwarding
ipv6 forwarding
!
!
line vty
!
```

- ospfd.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:59:50
```

```
!
hostname CAFE_62
password zebra
!
!
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
router ospf
 ospf router-id 10.203.62.229
 redistribute connected
 redistribute bgp
 network 10.203.62.224/28 area 0.0.0.1
!
line vty
!
```

- ospf6d.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:59:50
!
hostname ospf6d@CAFE_62
password zebra
!
debug ospf6 lsa unknown
!
interface eth1
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 10.203.62.229
 redistribute connected
 interface eth1 area 0.0.0.1
!
line vty
!
```

A.2.10.9 CAFE_G Configuration

- zebra.conf

```
!
```

```
! Zebra configuration saved from vty
!   2012/04/10 18:53:25
!
hostname CAFE_G
password zebra
enable password zebra
!
interface dummy0
 ipv6 nd suppress-ra
!
interface eth0
 ipv6 nd suppress-ra
!
interface eth1
 link-detect
 ipv6 nd suppress-ra
!
interface eth2
 link-detect
 ipv6 nd suppress-ra
!
interface eth3
 link-detect
 ipv6 nd suppress-ra
!
interface lo
 ipv6 address 2001:67c:24cc:31e0::cafe:9/128
!
ip forwarding
ipv6 forwarding
!
!
line vty
!
```

- bgpd.conf

```
!
! Zebra configuration saved from vty
!   2012/04/10 18:53:25
!
hostname CAFE_G
password zebra
!
router bgp 63
 bgp router-id 10.203.63.225
 network 10.203.63.0/24
 neighbor 10.203.62.246 remote-as 62
 neighbor 10.203.62.246 ebgp-multihop 10
 neighbor 10.203.63.247 remote-as 61
 neighbor 10.203.63.247 ebgp-multihop 10
!
 address-family ipv6
 network 2001:67c:24cc:31e0::/60
 neighbor 10.203.62.246 activate
 neighbor 10.203.63.247 activate
 exit-address-family
```

```
!
line vty
!
```

A.3 PTL LOCAL TESTBED

A.3.1 Addressing Schema

Machine	Interface	IPv4	IPv6
cafe-ptlwut	eth0 tap1 eth1	217.27.59.122 192.167.17.1 10.50.50.2	 3105::2
cafe-ptltid	eth0 tap1 eth1	217.27.59.121 192.167.16.6 10.50.50.3	 3105::3
CREr	eth0	10.50.50.4	3105::4
CREa	eth0	10.50.50.5	3105::5
CME	eth0	10.50.50.6	3105::6
CP	eth0	10.50.50.7	3105::7
RAE	eth0	10.50.50.8	3105::8
SNME	eth0	10.50.50.9	3105::9
cafe-ptl3	eth0 eth1	10.50.50.3 101.101.101.10	3105::3 3101::10
CC_{PTL1}	eth0	101.101.101.2	3101::2
CC_{PTL2}	eth0	101.101.101.12	3101::12
CS_{PTL1}	eth0	101.101.101.3	3101::3
CS_{PTL2}	eth0	101.101.101.13	3101::13

Table 60: PTL Addressing Schema

A.3.2 Forwarding Keys

CAFE	Key Name	Interface (device)	Destination MAC Address	Remote cafe associated	Remote interface
cafe-ptl3	0x01	eth0	00:0c:29:18:4a:a2	cafe-pt l3	eth0
cafe-pt l3	0x21	eth0	00:0c:29:34:f0:0a	cafe-ptlwut	eth1
cafe-pt l3	0x31	eth0	00:0c:29:63:4f:f8	cafe-ptltid	eth1
cafe-ptlwut	0x22	eth1	00:0c:29:18:4a:a2	cafe-pt l3	eth0
cafe-ptltid	0x32	eth1	00:0c:29:18:4a:a2	cafe-pt l3	eth0
cafe-ptlwut	0x11	eth1	00:0c:29:63:4f:f8	cafe-ptltid	eth1
cafe-ptltid	0x12	eth1	00:0c:29:34:f0:0a	cafe-ptlwut	eth1
cafe-ptlwut	0xb1	eth0	08:00:27:e0:c2:46	cafe-d (wut)	eth2
cafe-ptltid	0xc1	eth0	de:58:75:73:c8:f3	CAFE-TIDpt (tid)	br1

Table 61: PTL Forwarding Keys

A.3.3 IP Prefixes/CAFE Mapping

Prefix	Prefix Length	Edge Cafe
101.101.101.0	24	10.50.50.10

Table 62: PTL Subnet/CAFE Mapping in IPv4

Prefix	Prefix Length	Edge Cafe
3101::0	64	3101::10

Table 63: PTL Subnet/CAFE Mapping in IPv6

A.3.4 Peering AS/CAFE Mappings

AS Path	CoS	Source CAFE	Sink CAFE	Peer CME	BW
41,61	BTBE	10.50.50.2	10.203.61.227	10.203.61.228	100MB
41,61	BTBE	10.50.50.2	10.203.61.227	10.203.61.228	50MB
61,41	PR	10.203.61.227	10.50.50.2	10.203.61.228	100MB
61,41	BTBE	10.203.61.227	10.50.50.2	10.203.61.228	50MB
51,41	PR	10.95.51.81	10.50.50.3	10.95.51.3	100MB
51,41	BTBE	10.95.51.81	10.50.50.3	10.95.51.3	50MB
41,51	PR	10.50.50.3	10.95.51.81	10.95.51.3	100MB
41,51	BTBE	10.50.50.3	10.95.51.81	10.95.51.3	50MB
51,62	PR	10.95.51.85	10.203.62.227	10.203.62.228	100MB

Table 64: PTL Peering AS/CAFE Mappings for IPv4

AS Path	CoS	Source CAFE	Sink CAFE	Peer CME	BW
41,61	BTBE	3105::2	2001:67c:24cc:31c0:0:0:cafe:d	2001:67c:24cc:31c1:0:0:ae:1	100MB
41,61	BTBE	3105::2	2001:67c:24cc:31c0:0:0:cafe:d	2001:67c:24cc:31c1:0:0:ae:1	50MB
61,41	PR	2001:67c:24cc:31c0:0:0:cafe:d	3105::2	2001:67c:24cc:31c1:0:0:ae:1	100MB
61,41	BTBE	2001:67c:24cc:31c0:0:0:cafe:d	3105::2	2001:67c:24cc:31c1:0:0:ae:1	50MB
51,41	PR	2a02:9008:0:1918:0:50:56a3:5d	3105::3	2a02:9008:0:1911:0:50:56a3:50	100MB
51,41	BTBE	2a02:9008:0:1918:0:50:56a3:5d	3105::3	2a02:9008:0:1911:0:50:56a3:50	50MB
41,51	PR	3105::3	2a02:9008:0:1918:0:50:56a3:5d	2a02:9008:0:1911:0:50:56a3:50	100MB
41,51	BTBE	3105::3	2a02:9008:0:1918:0:50:56a3:5d	2a02:9008:0:1911:0:50:56a3:50	50MB
51,62	PR	2a02:9008:0:1918:0:50:56a3:5d	2001:67c:24cc:31c0:0:0:cafe:d	2001:67c:24cc:31c1:0:0:ae:1	100MB

Table 65: PTL Peering AS/CAFE Mappings for IPv6

A.3.5 Peering CAFE/Key Mappings

Type	$CAFE_{or} \rightarrow CAFE_{dest}$	Key list (reverse order)	CoS	BW
TRANSIT	10.50.50.10 → 10.50.50.10	0x01	PR	1GB
TRANSIT	10.50.50.10 → 10.50.50.2	0x21	PR	1GB
TRANSIT	10.50.50.10 → 10.50.50.3	0x31	PR	1GB
TRANSIT	10.50.50.2 → 10.50.50.10	0x22	PR	1GB
TRANSIT	10.50.50.3→ 10.50.50.10	0x32	PR	1GB
TRANSIT	10.50.50.2 → 10.50.50.3	0x11	PR	1GB
TRANSIT	10.50.50.3→ 10.50.50.2	0x12	PR	1GB
PEERING	10.50.50.2→ 10.203.61.227	0xb1	PR	100MB
PEERING	10.50.50.3→ 10.95.51.81	0xc1	PR	100MB
TRANSIT	10.50.50.10 → 10.50.50.10	0x0a	PR	500MB
TRANSIT	10.50.50.10 → 10.50.50.2	0x2a	PR	500MB
TRANSIT	10.50.50.10 → 10.50.50.3	0x3a	PR	500MB
TRANSIT	10.50.50.2 → 10.50.50.10	0x2b	PR	500MB
TRANSIT	10.50.50.3→ 10.50.50.10	0x3b	PR	500MB
TRANSIT	10.50.50.2 → 10.50.50.3	0x1a	PR	500MB
TRANSIT	10.50.50.3→ 10.50.50.2	0x1b	PR	500MB
PEERING	10.50.50.2→ 10.203.61.227	0xba	PR	50MB
PEERING	10.50.50.3→ 10.95.51.81	0xca	PR	50MB

Table 66: PTL Peering CAFEs/Keys Mappings in IPv4

Type	$CAFE_{or} \rightarrow CAFE_{dest}$	Key list (reverse order)	CoS	BW
TRANSIT	3105::10 →3105::10	0x01	PR	1GB

Type	$CAFE_{or} \rightarrow CAFE_{dest}$	Key list (reverse order)	CoS	BW
TRANSIT	3105::10 →3105::2	0x21	PR	1GB
TRANSIT	3105::10 →3105::3	0x31	PR	1GB
TRANSIT	3105::2 →3105::10	0x22	PR	1GB
TRANSIT	3105::3→3105::10	0x32	PR	1GB
TRANSIT	3105::2 →3105::3	0x11	PR	1GB
TRANSIT	3105::3→3105::2	0x12	PR	1GB
PEERING	3105::2→2001:67c:24cc:31c0:0:0:cafe:d	0xb1	PR	100MB
PEERING	3105::3→2a02:9008:0:1918:0:50:56a3:5d	0xc1	PR	100MB
TRANSIT	3105::10 →3105::10	0x01	PR	500MB
TRANSIT	3105::10 →3105::2	0x21	PR	500MB
TRANSIT	3105::10 →3105::3	0x31	PR	500MB
TRANSIT	3105::2 →3105::10	0x22	PR	500MB
TRANSIT	3105::3→3105::10	0x32	PR	500MB
TRANSIT	3105::2 →3105::3	0x11	PR	500MB
TRANSIT	3105::3→3105::2	0x12	PR	500MB
PEERING	3105::2→2001:67c:24cc:31c0:0:0:cafe:d	0xba	PR	50MB
PEERING	3105::3→2a02:9008:0:1918:0:50:56a3:5d	0xca	PR	50MB

Table 67: PTL Peering CAFEs/Keys Mappings in IPv6

A.3.6 Content Client to COMET CoS Mappings

Host	Client IP	CoS
CC_{PTL1}	101.101.101.2	PR
CC_{PTL2}	101.101.101.12	BE

Table 68: PTL CC to CoS Mapping in IPv4

Host	Client IP	CoS
CC_{PTL1}	3101::2	PR
CC_{PTL2}	3101::12	BE

Table 69: PTL CC to CoS Mapping in IPv6

A.3.7 Content Names

Name	CoS	QoS	URL	MIME	CME
ptl3.cy /adSD	PR	BW:5 Delay: 0.4 Loss: 0.01	http://101.101.101.3:8008/video.ts http://101.101.101.13:8008/video.ts	stream/mpeg	10.50.50.6
ptl3.cy/adSD_BtBE	BtBE	BW:5 Delay: 0.4 Loss: 0.01	http://101.101.101.3:8080/trailer.ts http://101.101.101.13:8080/trailer.ts	stream/mpeg	10.50.50.6
ptl3.cy/adSD_BE	BE	BW:5 Delay: 0.4 Loss: 0.01	http://101.101.101.3:8080/trialer.ts http://101.101.101.13:8080/trailer.ts	stream/mpeg	10.50.50.6
ptl3.cy/adVOD	PR	BW:10 Delay: 0,4 Loss: 0.001	http://101.101.101.3/video.ts http://101.101.101.13/video.ts	stream/mpeg	10.50.50.6
ptl3.cy /adVOD_BtBE	BtBE	BW:10 Delay: 0,4 Loss: 0.001	http://101.101.101.3/trailer.ts http://101.101.101.13/trailer.ts	stream/mpeg	10.50.50.6
ptl3.cy /adVOD_BE	BE	BW:10 Delay: 0,4 Loss: 0.001	http://101.101.101.3/trailer.ts http://101.101.101.13/trailer.ts	stream/mpeg	10.50.50.6

Table 70: PTL Content Names defined for IPv4

Name	CoS	QoS	URL	MIME	CME
ptl3.cy /adSD	PR	BW:5 Delay: 0.4 Loss: 0.01	http://[3101::3]:8008/video.ts http://[3101::13]:8008/video.ts	stream/mpeg	3105::6
ptl3.cy/adSD_BtBE	BtBE	BW:5 Delay: 0.4 Loss: 0.01	http://[3101::3]:8080/trailer.ts http://[3101::13]:8080/trailer.ts	stream/mpeg	3105::6
ptl3.cy/adSD_BE	BE	BW:5 Delay: 0.4 Loss: 0.01	http://[3101::3]:8080/trialer.ts http://[3101::13]:8080/trailer.ts	stream/mpeg	3105::6
ptl3.cy/adVOD	PR	BW:10 Delay: 0,4 Loss: 0.001	http://[3101::3]/video.ts http://[3101::13]/video.ts	stream/mpeg	3105::6
ptl3.cy /adVOD_BtBE	BtBE	BW:10 Delay: 0,4 Loss: 0.001	http://[3101::3]/trailer.ts http://[3101::13]/trailer.ts	stream/mpeg	3105::6

ptl3.cy /adVOD_BE	BE	BW:10 Delay: 0,4 Loss: 0.001	http://[3101::3]/trailer.ts http://[3101::13]/trailer.ts	stream/mpeg	3105::6

Table 71: PTL Content Names defined for IPv6

A.3.8 OpenVPN Configuration

A.3.8.1 Configuration file of OpenVPN client at CAFE-PTLTIDfor establishingTID-PTL tunnel

```
# This VPN is used for COMET to connect to TID server
client
dev tap
lladdr DE:17:90:8E:D0:8F
proto udp
remote  130.206.214.70 2011
persist-tun

#certificates and encryption
ca   /opt/etc/OpenVPN/ca.crt
cert /opt/etc/OpenVPN/pt-tid.crt
key  /opt/etc/OpenVPN/pt-tid.key

cipher none
comp-lzo
```

A.3.8.2 Configuration file of OpenVPN server at CAFE-PTLWUT for enabling PTL-WUT tunnel

```
#This VPN is used for COMET to enable connection of WUT to PTL
local 217.27.59.123
port 2011
proto udp
dev tap1
lladdr 3A:45:A2:CD:5E:46

#certificates and encryption
ca /opt/etc/OpenVPN/easy-rsa/keys/ca.crt
cert /opt/etc/OpenVPN/easy-rsa/keys/server.crt
key /opt/etc/OpenVPN/easy-rsa/keys/server.key # This file should be kept secret
dh /opt/etc/OpenVPN/easy-rsa/keys/dh1024.pem

persist-tun
mode server
tls-server
```

```
ifconfig 192.168.17.1 255.255.255.0
ifconfig-pool 192.168.17.2 192.168.17.10 255.255.255.0
keepalive 10 120
cipher none
comp-lzo
```

A.3.9 Quagga Configuration

A.3.9.1 CAFE_PTLTID Configuration

- zebra.conf

```
! Zebra configuration saved from vty
!   2012/10/24 18:30:56
!
password zebra
enable password zebra
!
interface dummy0
ipv6 nd suppress-ra
!
interface eth0
 no ipv6 nd suppress-ra
!
interface eth1
 ipv6 address 3105::3/64
 no ipv6 nd suppress-ra
!
interface eth2
 no ipv6 nd suppress-ra
!
interface lo
!
interface tap0
 no ipv6 nd suppress-ra
!
ip route 192.168.17.0/24 10.50.50.2
!
ip forwarding
ipv6 forwarding
```

```
!
!
line vty
!
```

- bgpd.conf

```
! Zebra configuration saved from vty
!   2012/10/24 18:30:56
!
password zebra
!
router bgp 41
 bgp router-id 10.50.50.3
 network 10.50.50.0/24
 network 101.101.101.0/24
 neighbor 10.50.50.2 remote-as 41
 neighbor 192.168.16.1 remote-as 51
 neighbor 192.168.16.1 ebgp-multihop 10
!
 address-family ipv6
 network 3101::/64
 network 3105::/64
 neighbor 10.50.50.2 activate
 neighbor 10.50.50.2 next-hop-self
 neighbor 192.168.16.1 activate
 exit-address-family
!
access-list all permit any
!
line vty
!
```

- ospfd.conf

```
!
! Zebra configuration saved from vty
!   2012/10/24 18:30:56
!
```

```
password zebra
!
!
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
interface tap0
!
router ospf
 ospf router-id 10.50.50.32
 network 10.50.50.0/24 area 0.0.0.0
!
line vty
!
```

- ospf6d.conf

```
!
! Zebra configuration saved from vty
!   2012/10/24 18:30:56
!
password zebra
!
debug ospf6 lsa unknown
!
interface eth1
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
```

```
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 10.50.50.3
 interface eth1 area 0.0.0.0
!
line vty
!
```

A.3.9.2 CAFE_PTLWUT Configuration

- zebra.conf

```
!
! Zebra configuration saved from vty
!   2012/10/24 18:30:49
!
password zebra
enable password zebra
!
interface dummy0
 ipv6 nd suppress-ra
!
interface eth0
 no ipv6 nd suppress-ra
!
interface eth1
 ipv6 address 3105::2/64
 no ipv6 nd suppress-ra
!
interface eth2
 no ipv6 nd suppress-ra
!
interface lo
!
interface tap1
 no ipv6 nd suppress-ra
```

```
!
ip route 192.168.16.0/24 10.50.50.3
!
ip forwarding
ipv6 forwarding
!
!
line vty
!
```

- bgpd.conf

```
!
! Zebra configuration saved from vty
!   2012/10/24 18:30:49
!
password zebra
!
router bgp 41
 bgp router-id 10.50.50.2
 network 10.50.50.0/24
 network 101.101.101.0/24
 neighbor 10.50.50.3 remote-as 41
 neighbor 192.168.17.2 remote-as 61
 neighbor 192.168.17.2 ebgp-multihop 10
!
 address-family ipv6
 network 3101::/64
 network 3105::/64
 neighbor 10.50.50.3 activate
 neighbor 10.50.50.3 next-hop-self
 neighbor 192.168.17.2 activate
 exit-address-family
!
access-list all permit any
!
line vty
!
```

- ospfd.conf

```
!
! Zebra configuration saved from vty
!   2012/10/24 18:30:49
!
password zebra
!
!
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
interface tap1
!
router ospf
 ospf router-id 10.50.50.22
 redistribute bgp
 network 10.50.50.0/24 area 0.0.0.0
!
line vty
!
```

- ospf6d.conf

```
!
! Zebra configuration saved from vty
!   2012/10/24 18:30:49
!
password zebra
!
debug ospf6 lsa unknown
!
interface eth1
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
```

```
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 10.50.50.2
 redistribute bgp
 interface eth1 area 0.0.0.0
!
line vty
!
```

A.3.9.3 CAFE_PTL3 Configuration

- zebra.conf

```
! Zebra configuration saved from vty
!   2012/10/24 18:33:04
!
password zebra
!
interface dummy0
 ipv6 nd suppress-ra
!
interface eth0
 link-detect
 ipv6 address 3105::10/64
 no ipv6 nd suppress-ra
!
interface eth1
 link-detect
 ipv6 address 3101::10/64
 no ipv6 nd suppress-ra
!
interface eth2
 ipv6 nd suppress-ra
!
interface lo
!
ip forwarding
ipv6 forwarding
```

```
!
!
line vty
!
```

- ospfd.conf

```
!
! Zebra configuration saved from vty
!   2012/10/24 18:33:04
!
password zebra
!
!
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
router ospf
 ospf router-id 10.50.50.102
 redistribute connected
 redistribute bgp
 network 10.50.50.0/24 area 0.0.0.0
 network 101.101.101.0/24 area 0.0.0.1
!
line vty
!
```

- ospf6d.conf

```
!
! Zebra configuration saved from vty
!   2012/10/24 18:33:04
```

```
!
password zebra
!
debug ospf6 lsa unknown
!
interface eth0
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
interface eth1
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
router ospf6
 router-id 10.50.50.10
 redistribute connected
 interface eth0 area 0.0.0.0
 interface eth1 area 0.0.0.1
!
line vty
!
```

A.4 Use Cases Configuration

Name	CoS	QOS	URL	MIME	CME
tid1.es/demo_a	PR	BW:25 Delay: 0,6 Loss: 10	http://[2a02:9008:0:1913:0:50:56a3:62]:8008/cleo.ts http://[2a02:9008:0:1912:0:50:56a3:48]:8008/cleo.ts	stream/mpeg	2a02:9008:0:1911:0:50:56a3:4e
tid1.es/demo_b	PR	BW:25 Delay: 0,6 Loss: 10 Priority 1	http://[2a02:9008:0:1913:0:50:56a3:62]:8008/cleo.ts	stream/mpeg	2a02:9008:0:1911:0:50:56a3:4e
	PR	BW:25 Delay: 0,4 Loss: 0,001 Priority 2	http://[3101::3]:8008/video_HQ.ts	stream/mpeg	3105:6
	BTBE	BW:25 Delay: 0,6 Loss: 10 Priority 1	http://[3101::3]:8080/video_LQ.ts	stream/mpeg	3105:6
	BE	BW:25 Delay: 0,6 Loss: 10 Priority 1	http://[2001:67c:24cc:31d3::242]:8888/get/127.ts	stream/mpeg	2001:67c:24cc:31d1::ae:2
tid1.es/demo_c	PR	BW:25 Delay: 0,6 Loss: 10 Priority 1	http://[2a02:9008:0:1913:0:50:56a3:62]/videos/CLEOPATRA_DISC_1_480p.ts	stream/mpeg	2a02:9008:0:1911:0:50:56a3:4e
tid1.es/demo_P2P	PR	BW:25 Delay: 0,6 Loss: 10	PR ttp://[2a02:9008:0:1913:0:50:56a3:62]:8008/cleo.ts	stream/mpeg	2a02:9008:0:1911:0:50:56a3:4e
	BE	BW:25 Delay: 0,6 Loss: 10	BE http://[2a02:9008:0:1912:0:50:56a3:4a]/media/cleoP2P.torrent	text/torrent	2a02:9008:0:1911:0:50:56a3:4e

Table 72: Content Names for use cases

8 References

[1] COMETDeliverable D2.1Business Models and System Requirements for the COMET System

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

[3] COMET Deliverable D3.2Final Specification of Mechanisms, Protocols and Algorithms for the Content Mediation System

[4] COMET Deliverable D4.2 Final Specifications of the Mechanisms, Protocols and Algorithms for Enhanced Network Platforms

[5] COMET Deliverable D5.1 Integration of COMET Prototype and Adaptation of Applications

[6] COMET Deliverable D5.2 Scalability of COMET System

[7] COMET Deliverable D6.1 Demonstration Scenarios and Test Plan

[8] COMET Deliverable D4.3 Prototype Implementation and System Integration Interfaces for Enhanced Network Platforms

[9] The Cooperative Association for Internet Data Analysis, http://www.caida.org/.

[10] University of Oregon Route Views Archive Project David Meyer, http://archive.routeviews.org/.

[11] RIPE Network Coordination Centre, “Routing Information Service”, http://www.ripe.net/data-tools/stats/ris/ris-peering-policy.

[12] Craig Labovitz, Abha Ahuja, Abhijit Bose, and Farnam Jahanian,“Delayed Internet routing convergence”, SIGCOMM Comput. Commun. Rev. 30, 4, August 2000, pp. 175-187.

[13] Bradner, S. and J. McQuaid, "Benchmarking Methodology for etwork Interconnect Devices", RFC 2544, March 1999.

[14] http://www.videolan.org/

[15] http://www.utorrent.com/intl/es/

[16] http://www.envision-project.org/

[17] http://openvpn.net/

9 Abbreviations

AS	Autonomous System
BE	Best Effort
BGP	Border Gateway Protocol
BtBE	Better than Best Effort CoS
BW	Bandwidth
CAFE	Content-Aware Forwarding Entity
CINA	Collaboration Interface between Network and Application
CC	Content Client
CFP	Content Forwarding Plane
CME	Content Mediation Entity
CMP	Content Mediation Plane
COMET	COntent Mediator architecture for content-aware nETworks
CoS	Class of Service
CP	Content Publisher
CRE	Content Resolucion Entity
CRL	Content Retrieval Latency
CRSR	Content Retrieval Success Ratio
CRT	Content Resolution Time
CS	Content Server
CSR	Content Streaming Relay
DL	Direct Download
ENVISION	Co-optimisation of overlay applications and underlying networks
FIB	Forwarding Information Base
HTTP	HyperText Transfer Protocol
IP	Internet Protocol
IPLR	IP Packet Loss Ratio
IPTD	IP Packet Transfer Delay
ISP	Internet Service Provider
LAN	Local Area Network
P2P	Peer to Peer
PKI	Public Key Infrastructure
OSPF	Open Shortest Path First
Pr	Premium CoS
PTL	PrimeTel
QoS	Quality of Service
RAE	Route Awareness Entity

RIP	Routing Information Protocol
SIC	Server Information Collector
SLA	Service Level Agreement
SNME	Server and Network Monitoring Element
STREP	Specific Targeted Research Project
TID	Telefonica I+D
URL	Universal Resource Locator
VLAN	Virtual LAN
VLC	VideoLan Player
VoD	Video On Demand
VPN	Virtual Private Network
WUT	Warsaw University of Technology

10 Acknowledgements

This deliverable was made possible due to the large and open help of the WP5 team of the COMET team within this STREP. Many thanks to all of them.