Emulation of Open Content Aware Networks on the iLab.t Virtual Wall

Steven Latré, Jeroen Famaey, Wim Van de Meerssche,
Tim Wauters, and Filip De Turck

Ghent University - IBBT, Department of Information Technology, Belgium
Steven.Latre@intec.ugent.be

Abstract. The FP7 OCEAN project investigates scalable content and network mechanisms to deliver multimedia services with high quality over a Content Delivery Network (CDN) deployed in the last mile. To evaluate the solutions proposed by OCEAN, a three step approach was taken in which the solutions were first simulated, later emulated and ultimately deployed in a field trial. In this article, we focus on the emulation of OCEAN components on the iLab.t Virtual Wall. We discuss the benefit of performing emulations, both for the simulations and field trial, through some specific illustrative OCEAN examples.

1 Introduction

The increasing popularity of multimedia services with stringent quality requirements, such as catch-up TV and Video on Demand, has given rise to the need for multimedia-optimized delivery mechanisms. Content Delivery Networks (CDNs) are an essential part in this for the optimized distribution of the associated content. However, the current CDN solutions focus mainly on optimizing the delivery in the regional and core networks. As illustrated in Figure 1, the FP7 OCEAN project investigates the deployment of CDN solutions to optimize multimedia delivery on the last mile. More specifically, OCEAN focuses on (i) the deployment of caches deeper into the aggregation networks, up to the access nodes and (ii) enabling network-controlled adaptive content delivery (e.g., through HTTP adaptive streaming techniques).

To design and evaluate these OCEAN solutions, a three step approach is taken. In a first phase, a wide range of novel network functions (e.g., video-optimized caching algorithms and admission control mechanisms enabling adaptive content delivery) were simulated independently of other OCEAN contributions. This allowed doing an elaborate parameter sweep and identifying the best performing solutions for the remainder of the project. In a second phase, a subset of the, in the first phase, proposed solutions were integrated and emulated on the iLab.t Virtual Wall testbed facility. These emulations serve a double goal: on one hand they allow confirming (or contradicting) the simulation-based experiments by evaluating their performance under more realistic conditions. On the other hand, the emulations allow preparing the software prototypes for a

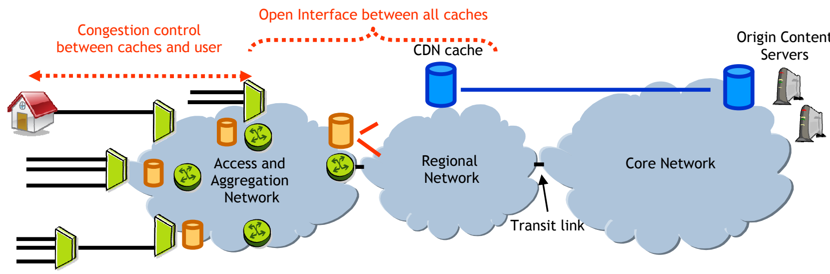


Fig. 1: Overview of the OCEAN architecture.

large-scale evaluation in a field trial, which is the third phase of the OCEAN experimentation approach. In this article, we discuss how the emulations were carried out.

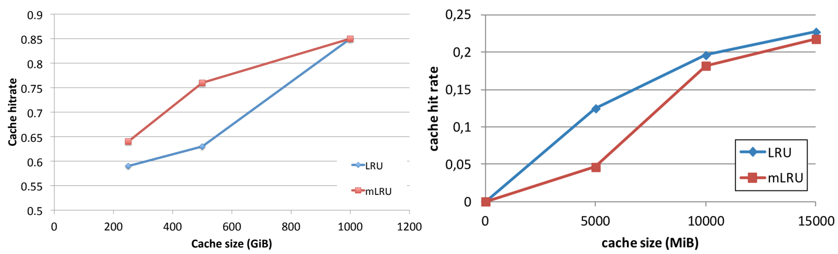
2 Emulation facility

The OCEAN emulation experiments are run on top of the iLab.t Virtual Wall testbed facility, which is a large-scale network emulation environment that consists of 100 nodes interconnected via a non-blocking 1.5 Tbps VLAN Ethernet switch, and a display wall (20 monitors) for experiment visualization. Each server is connected with either 4 or 6 gigabit Ethernet links to the switch (of which one serves as a control interface). The Virtual Wall nodes can be assigned different functionalities ranging from terminal, server and network node to impairment node. The management of the physical topology is carried out by the Emulab testbed management software.

Furthermore, specifically for the OCEAN experiments, an automation tool has been developed, called Wall Experiment Organizer (WExO). WExO alleviates the problems associated with manual experiment setup on the Virtual Wall testbed. WExO allows defining planned experiments and parameter sweeps via an XML file or by using a graphical user interface. It uses a set of plugins to define the required emulation components. The functionality offered by these plugins ranges from dynamic traffic shaping to emulation of end-user behaviour and monitoring experiment metrics. Additionally, WExO supports straightforward repeatability of experiments and automated log and result processing.

3 Lessons learned through emulation

Within the OCEAN experimentation strategy, the emulations were carried out after the majority of the simulation experiments and before the field trial. Therefore, the results obtained through emulation were able to suggest improvements for additional experiments in the simulation phase as well as improve the devised solutions for deployment in the field trial. In this section, we briefly describe the main advantages of the emulations and how they have helped the experimentation in the other phases.



(a) Simulation (b) Emulation

Fig. 2: Comparison of cache algorithm performance obtained through simulation (a) and emulation (b).

Performance evaluation under more realistic scenarios In OCEAN's simulation phase, several simplifying assumptions were made that allowed speeding up the time required to perform the simulations (thus effectively allowing to perform more detailed simulations), with the cost of a loss in realism. Once simulation found a good parameter setting for a particular algorithm, this setting was evaluated through emulation to confirm the obtained simulation results. In some cases, the emulation results showed significant differences due to the greater accuracy in the modelled environment (e.g., more bursty traffic). This allowed providing suggestions for improvements for the simulation environment and algorithmic contributions.

Integrated evaluation of OCEAN solutions During simulation, each OCEAN solution was evaluated independently of the other. In the emulations, the different solutions were integrated and their combined performance was evaluated. In some cases, this provided some important insights. This is illustrated in Figure 2, which shows the performance of the first version of a novel caching algorithm (i.e., modified LRU – mLRU) compared to the traditional LRU approach. While initial simulation-based results showed that mLRU performed best, the emulations indicated the contrary. This was because, in the emulations, it was combined with work on HTTP Adaptive Streaming (HAS). The use of HAS negatively influenced mLRU performance. These conclusions were used to suggest improvements to the design and configuration of mLRU.

Prototype testing for field trial deployment Finally, in order to adequately prepare the field trial, several prototypes were first evaluated through emulation. In these experiments, the focus was on carrying out stress tests to identify possible bottlenecks in the software. As a result, several issues (e.g., memory leaks) were identified, which were resolved before the field trial phase was initiated.

Acknowledgement The research leading to these results has received funding from the European Union's Seventh Framework Programme ([FP7/2007-2013]) under grant agreement n 248775.