MediaCloud

a Framework for Real-time Media Processing in the Network

Peter Domschitz, Markus Bauer
Alcatel-Lucent, Bell Labs, Stuttgart, Germany
{Peter.Domschitz, Markus.Bauer}@alcatel-lucent.com

I. MOTIVATION

Predicted growth of media traffic (even conservative forecasts predict an increase of 40% year over year) can only give a hint of how important video will be for future entertainment, business, and government services. But such future multimedia services will be different from what video means to all of us today. More and more people will not only consume, but will also actively produce content. These services will be used by mobile users, and personalization of content will be the common case. Service introduction, enhancements, and adoption of services will be much more dynamic than today, calling for flexible Software as a Service business and operational models.

II. STATE OF THE ART

Up to now ever increasing Internet capacity demands could be accommodated by simply enhancing the installed bandwidth. As this concept becomes more and more challenging moving forward [1], there is need for alternative approaches. A standard practice to achieve this goal is to add "higher layer" intelligence to the networks aiming to reduce overall traffic, thus enhancing available capacity by reducing traffic. The first huge success of this concept was the invention and successful introduction of Content Delivery Networks. CDNs basically enabled the massive scale introduction of media services comprising broadcast delivery characteristics in the Internet.

At the same time, empowered by mature virtualization technologies, cloud computing infrastructures and services are going to change the IT world and its business models. Basically, cloud computing allows hosting services much more economically, by taking advantage of dynamic use of shared resources. This is a clear advantage for web and enterprise services. But beyond that simple transaction-oriented services there are other comparatively challenging services, which are time-sensitive and session-oriented, like multimedia.

III. MEDIA IS CHALLENGING

Currently we see an emerging trend towards personalized media streams, where media processing is performed somewhere in the Internet, i.e. in the cloud, enabling the evolution of, e.g., IP-TV towards multi-view video or cloud based gaming services. As CDNs are build for the efficient delivery of the same content to a multitude of receivers, individualized content streams are going to challenge the networks, because cache-assisted delivery schemes become less and less efficient. CDNs also fail to serve the demand of future media services with respect to user-specific stream processing [2].

Proliferation of individual content streams jeopardizes the whole Internet infrastructure.

Likewise today's 'Big Iron' approach to cloud computing leveraging large centralized data centers fails due to the fact that intermediate processing of media in the network, as ultimately required by personalized video services, cannot be provided at appropriate locations. The networks would not be able to economically scale with the inevitable traffic demand provoked by performing all required media processing in a limited number of large data centers.

Today's approach to cloud computing fails to provide the required levels of performance, dy2namicity and programmability.

Thus future media services call for huge amounts of networking and network based / cloud processing resources. Processing on millions of real-time streams implies requirements hardly be provided by today's operating systems, having its seeds in batch processing. Hence this is also the case for state of the art cloud infrastructures.

It's still an ongoing debate on what the right hardware technology is, e.g. GPUs vs. many-core-CPUs, to meet required efficiency and performance requirements. To install the proper hardware is just the necessary prerequisite to provide a usable cloud resource incorporated into a software framework. In order to select the right technology, it's crucial to understand the requirements of relevant media applications and the needed media processing components operating on the media flows.

In principle, heavily investing into networking and processing resources and spending huge amounts of developing resources into building media services can solve these issues. However,

large scale introduction and adoption of future media services will not happen unless there are much more efficient and economic ways of enabling the networks / the cloud to support the needs of future media services.

IV. THE MEDIACLOUD APPROACH

The G-Lab project NETCOMP provides an answer how to overcome those challenges by meeting future resource demands through distributed processing: With MediaCloud, Bell Labs provides a disruptive platform as a service approach for building and executing future media services.

Offload networks by localizing traffic.

Today's applications are typically designed in a way, that to be processed data is moved through the network to where the application is executed. We believe that this paradigm will change in the future, meaning that an intelligent infrastructure will optimize resource usage and user experience by moving applications or parts of it next to the data. Therefore we are working on ways to optimize the delivery of (real-time) media services on top of a distributed cloud environment. In fact, dynamic utilization of processing resources in multiple locations is the key to meet performance expectations and limit local resource bottlenecks.

In parallel there is a clear trend towards complex media flows and media mashups. This can be seen as a similar paradigm shift at the application layer. Both go hand in hand, resulting in a strong statement for distribution.

Enable the cloud to be the platform for building and executing future media services.

Moving from a centralized towards a distributed service execution paradigm requires software to be split up into service components. Composing a service from a plurality of components means building the required (atomic) service components and describing how those components interact, that is, which media flows components exchange at service runtime.

The developer of such a service does not need to care about specifying where in the network such components should be executed. MediaCloud will take care of finding best-fit resources during service runtime and the execution framework ensures fluent media flow forwarding between components.

Investigations and experiments have shown that using fully-fledged operating systems inside a virtual machine can hardly offer the required performance, scalability and efficiency for running distributed real-time media-centric services [2, 3].

Distributed execution environment for efficient processing on real-time flows.

MediaCloud aims to overcome those limitations [4]. It introduces a lightweight execution container design, which is fully optimized for supporting efficient execution of fine-grained service components. Basic framework mechanisms are fully tailored to the needs of distributed real-time flow processing. The framework supports service logic refinements at runtime, which means that service components can be added and deleted and media flows can be moved between, added to or removed from components at service runtime. Such dynamic mechanisms in combination with the ability to move service components between execution resources in the network during service runtime build the basic foundation for an efficient, top-performing and scalable service execution on distributed processing resources in the network.

Dynamic Runtime Placement of Service Components.

In today's cloud paradigm, a services execution resource is assigned prior to service runtime. Most often this mapping is a manual administrative task of deciding which data center should host a specific service. As a consequence, independent of the location of its origin, any data to be processed has to be sent towards its pre-assigned data center, be processed there and sent out to its destination. Apparently such a static mapping scheme does not allow improving service performance and resource utilization by dynamically adapting service execution at runtime. For example, allow a global video conferencing service to easily grow and shrink in number of served users, or move to more appropriate locations following the usage patterns, just by creating, taking down or moving necessary service components. This functionality must be provided by efficient means at platform level to avoid the need to build complex service logic (middleware) into each application taking care about service scaling and component locality.

For truly distributed service deployments, which will operate at least on the granularity of per-user, respectively per session components, educated component placement decisions can only be achieved during service runtime. Only at that time, sources and sinks of relevant media streams are known and thus media processing components can be assigned to close-by processing resources resulting in reduced end-to-end service latency and offloaded networks by keeping traffic local. Changing service profiles and varying traffic patterns might also ask for resource re-assignment during service runtime. Thus efficient component placement demands not only a one shot mapping decision when a component gets instantiated, but also requires ongoing evaluation of resource assignments.

The framework's lightweight component and execution container design provides the basic means to support the level of component agility such a dynamic resource assignment approach requires.

V. Outstanding Performance

First measurements performed on the prototype implementation proofed that MediaCloud is able to provide the envisaged level of agile resource allocation and utilization. It supports instantiation of media processing functions, i.e. service components on distributed cloud resources, in the timeframe of a few milliseconds. Even more, a re-assignment of media processing components from one processing resource onto another at service runtime is also achieved in the same timeframe of 2 to 3 milliseconds. This compares to highly optimized virtual machine based systems, which can achieve such tasks in seconds but not in milliseconds (Figure 1).

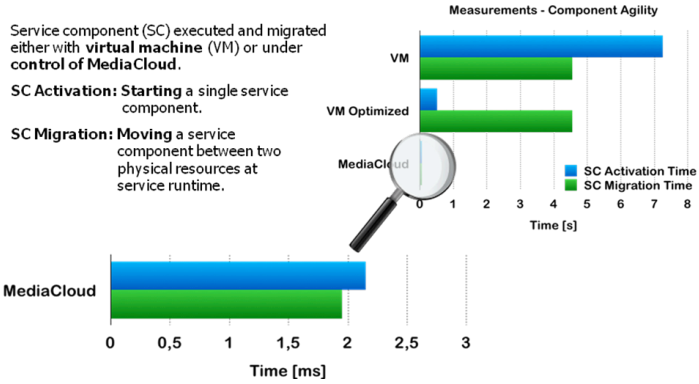

Figure 1. Agile Service Component Handling

Additional investigations indicate that MediaCloud is also able to achieve much more efficient resource utilization. A collection of cooperative media processing tasks executed on a MediaCloud controlled processing resource consumed only about half of the CPU cycles needed doing the same job by making each task a Linux process. At the same time we could show significantly better end-to-end service delay figures for a collection of media processing components executed on MediaCloud despite its more economical resource utilization.

VI. Conclusions

Future media services call for a new approach on how to build, deploy and execute media services on distributed cloud resources. By combining multimedia service delivery and networked cloud processing the G-Lab NETCOMP project follows a disruptive approach to provide cloud technology for media centric applications beyond today's Web services, even enabling services not possible today.

To achieve the required extreme levels of dynamicity, scaling, performance, and efficient resource utilization across distributed resources in the network, we introduce with MediaCloud a novel flow driven execution environment fully tailored to the needs of distributed real-time media processing. This extreme approach is fundamentally different to traditional cloud operating systems and their underlying cumbersome virtualization technologies.

In the talk we provide further insights into the MediaCloud technology. Demonstrations and performance measurements show the potential of the new concept.

[1] P. Domschitz, M. Bauer, J. Sienel and M. Kessler: "Move Applications not Data – A new Paradigm for the Future Internet" Proc. of the 10th Wuerzburg Workshop on IP, EuroView 2010.

[2] M. Bauer, S. Braun and P. Domschitz: "Media Processing in the Future Internet" Proc. of the 11th EuroView 2011.

[3] D. Kyriazis et. al.: "A Real-time Service Oriented Infrastructure" IADIS International Conference on Applied Computing, 2010.

[4] M. Bauer, S. Braun and P. Domschitz: "MediaCloud − A Distributed Service Platform for Media Services" KuVS 5th Workshop on Next Generation Service Delivery Platforms, NGSDP 2011.

This work is co-funded by the German Federal Ministry of Education and Research (BMBF) within the G-Lab project NETCOMP (FKZ 01BK0940).