Visualization of Dynamic Resource Allocation for HEVC Encoding in FPGA-Accelerated SDN Cloud

Panu Sjövall, Mikko Teuho, Arto Oinonen, Jarno Vanne, Timo D. Hämäläinen
Computing Sciences, Tampere University, Finland
{panu.sjovall, mikko.teuho, arto.oinonen jarno.vanne, timo.hamalainen}@tuni.fi

***Abstract*—This paper describes a demonstration setup to visualize dynamic resource allocation for real-time HEVC encoding services in FPGA-accelerated cloud. The demonstrated application is Kvazaar HEVC intra encoder, whose functionality is partitioned between FPGAs and processors. During the demonstration, several encoding services can be invoked with requests to the resource manager, which is responsible for allocation, deallocation, and load balancing of resources in the network. The manager provides JSON data to the visualizer, which uses D3 JavaScript library to visualize 1) the physical network structure; 2) running services; and 3) performance of the network elements. This interactive demonstration allows users to request new video streams, view the encoded streams, observe the visualization of the network and services, and manually turn on/off resources to test the robustness of the system.**

Keywords— Data Center processing, High Efficiency Video Coding (HEVC), Kvazaar HEVC encoder, Field-programmable gate array (FPGA), Software-defined networking (SDN)

I. Introduction

The rapidly increasing popularity and complexity of video coding, deep neural networks, and data analytics call for hardware acceleration in cloud computing. In the mainstream cloud computing systems, *field-programmable gate array* (*FPGA*) acceleration is typically implemented by PCIe cards attached to host servers [1], [2]. However, this approach ties the number of FPGAs to the server counts.

We solved this limitation by connecting FPGAs to servers via fiber and letting FPGAs act as independent nodes. We also replaced the full protocol stack implementations on the FPGAs with *Software-Defined Networking* (*SDN*). The SDN approach enables sharing any FPGA with any server through programmable data flows. The proposed system makes use of a proactive resource manager that dynamically switches between available software and hardware resources, without breaking up the live video stream.

II. Demonstration Setup

Fig. 1 shows our cloud architecture. It consists of three Xeon servers, two Intel Arria 10 FPGAs, and two HP SDN switches with HP VAN SDN Controller. The network components are specified in Table I.

Fig. 2 illustrates the demonstration setup. The prototype cloud is physically located at Tampere University (Fig. 2 (a)) and it is accessed over the network via VPN in the demonstration. A laptop (Fig. 2 (b)) is used for displaying the visualization interface, user interaction, and video playback.

The demonstrated application is Kvazaar HEVC encoder [3] which can be executed as a CPU-only service or it can be partitioned between CPUs and FPGA accelerators [4], [5]. Kvazaar is a standard software encoder [6] written in C and its hardware accelerator is implemented with Catapult-C high-level synthesis tool [7] from the C code. The inputs for all

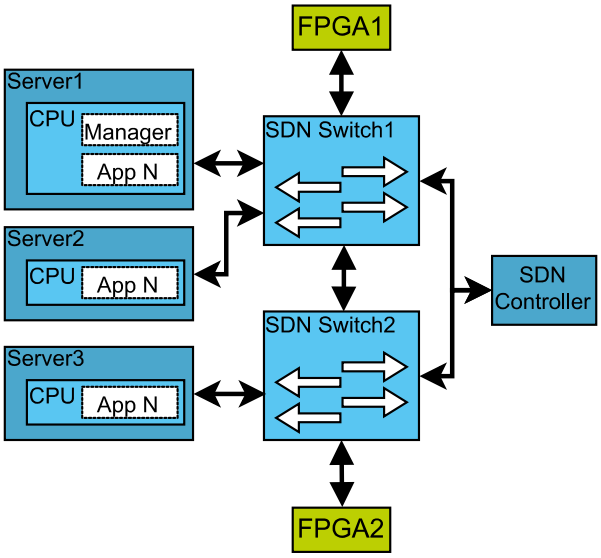

Fig. 1. Prototype cloud system

TABLE I. Cloud Component Specifications

Device	Type	CPU	Memory
Server1	HP Server	Xeon E5-2630	96GB
Server2	Nokia Airframe Cloud server [8]	Xeon E5-2680v4	256GB
Server3	Nokia Airframe Cloud server [8]	Xeon E5-2680v3	256GB
Switch1	HPE FlexFabric 5900AF 48G 4XG 2QSFP+	-	-
Switch2	HP Switch 5406Rzl2	-	-
FPGA1	Intel Arria 10 GX FPGA Dev Kit [9]	-	-
FPGA2	Intel Arria 10 GX FPGA Dev Kit [9]	-	-
Controller	HP VAN SDN Controller [10]	-	-

demonstrated encoding services are raw video files with different resolution. The output is in *HTTP Live Streaming* (*HLS*) format, which is decoded in live playback.

III. Run-Time Visualizer

The visualizer is written in JavaScript using D3 library. It gets input data in real time in JSON format from the manager. The visualizer illustrates dynamic deployment of HEVC encoding tasks in run time. It can work with an arbitrary set of resources and with varying number of encoding services.

The physical view of the network is shown in Fig. 2 (c). The symbols correspond to the device type and the server symbol size to the number of CPU cores. The physical view also shows 1) the CPU load graph inside the server symbol; 2) details as tooltips; and 3) connection bandwidth with changing line width. The services are shown in Fig. 2 (d) by dividing them as input sources, software (Kvazaar_HEVC) and hardware (Kvazaar_HEVC_acc) encoding services, and output destinations. Encoding speeds and bitrates are also shown for every service.

During the demonstration, existing computing resources can also be manually removed to see how the network self-organizes without breaking up video streaming. This can be seen from the visualization in real time.

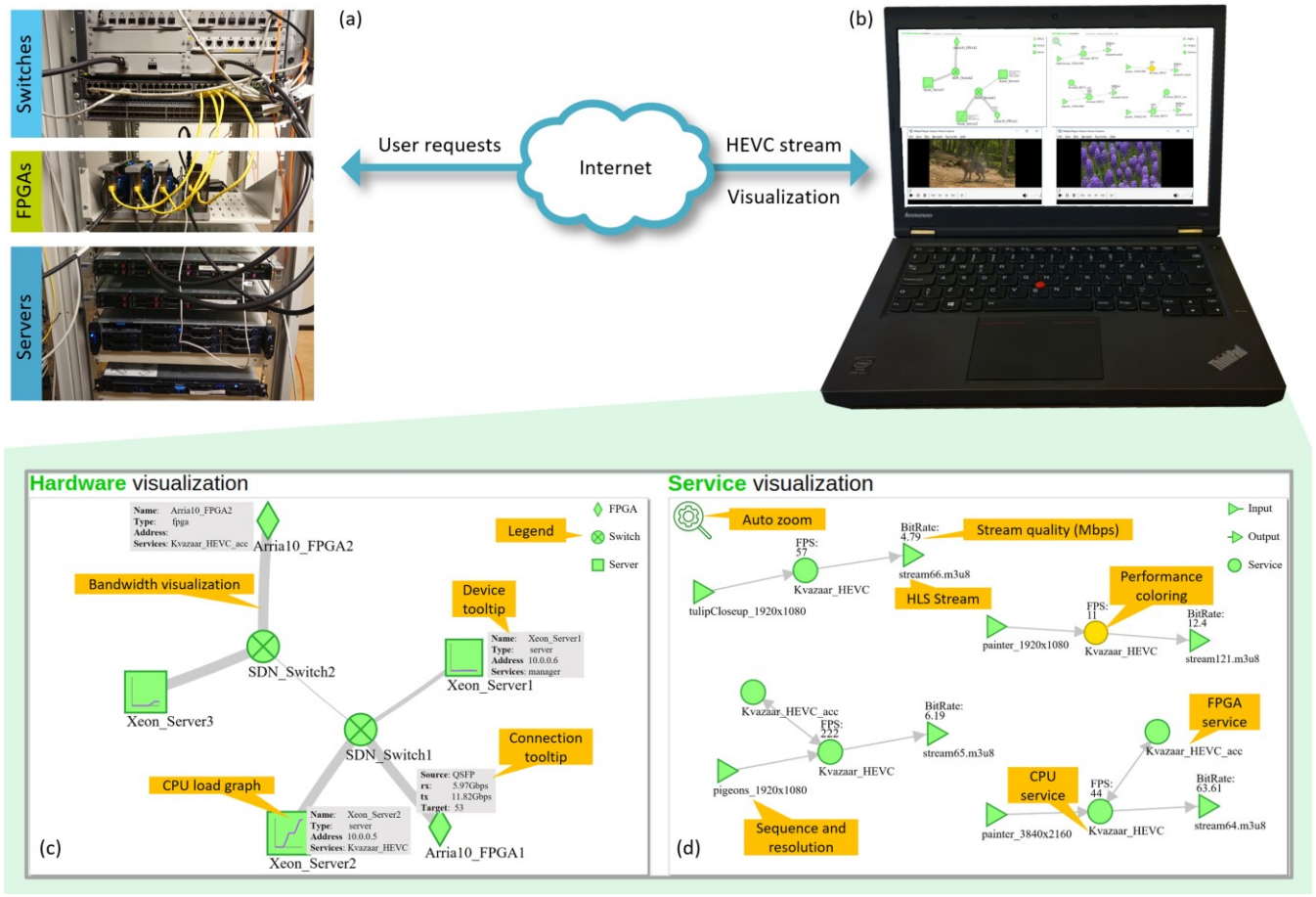

Fig. 2. The demonstration setup. (a) Remote server. (b) Laptop at the venue. (c) Physical view of the visualizer. (d) Service view of the visualizer.

IV. Conclusion

Our system with fiber connected FPGAs provides a new microservice based approach for hardware accelerated video encoding services in the cloud. This paper described a setup to demonstrate the basic operating principles of our proposal. In the demonstration, a special attention is paid to dynamic resource allocation for HEVC encoding services, switching service execution between CPUs and FPGAs, recovering from changes, and scalability of our architecture. The visualizer offers a real-time view of the available resources, running services, and performance.

Acknowledgment

This work was supported in part by the European Celtic-Plus project VIRTUOSE, the Academy of Finland (decision no. 301820), Nokia Foundation, and the Finnish Foundation for Technology Promotion.

References

[1] A. Putnam *et al.*, "A reconfigurable fabric for accelerating large-scale datacenter services," *IEEE Micro*, vol. 35, no. 3, May-June 2015, pp. 10-22.

[2] A. M. Caulfield *et al.*, "A cloud-scale acceleration architecture," *in Proc. Annual IEEE/ACM Int. Symp. Microarchitecture*, Taipei, Taiwan, Oct. 2016.

[3] Kvazaar HEVC encoder [Online]. Available: https://github.com/ultravideo/kvazaar

[4] P. Sjövall, V. Viitamäki, A. Oinonen, J. Vanne, T. D. Hämäläinen, and A. Kulmala, "Kvazaar 4K HEVC intra encoder on FPGA accelerated Airframe server," *in Proc. IEEE Workshop Signal Process. Syst.*, Lorient, France, Oct. 2017.

[5] P. Sjövall, V. Viitamäki, J. Vanne, T. D. Hämäläinen, and A. Kulmala, "FPGA-powered 4K120p HEVC intra encoder," *in Proc. IEEE Int. Symp. Circuits Syst.*, Florence, Italy, May 2018.

[6] M. Viitanen, A. Koivula, A. Lemmetti, A. Ylä-Outinen, J. Vanne, and T. D. Hämäläinen, "Kvazaar: open-source HEVC/H.265 encoder," *in Proc. ACM Int. Conf. Multimedia*, Amsterdam, The Netherlands, Oct. 2016.

[7] *Catapult High-Level Synthesis* [Online]. Available: https://www.mentor.com/hls-lp/catapult-high-level-synthesis/

[8] *AirFrame data center solution* [Online]. Available: https://networks.nokia.com/solutions/airframe-data-center-solution

[9] *Arria 10* [Online]. Available: https://www.intel.com/content/www/us/en/programmable/products/boards_and_kits/dev-kits/altera/kit-a10-gx-fpga.html

[10] *HP Virtual Application Networks SDN Controller* [Online]. Available: http://h17007.www1.hpe.com/docs/networking/solutions/sdn/4AA4-8807ENW.pdf