International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 02 | February - 2023 Impact Factor: 7.185 ISSN: 2582-3930

FPGA-based Solution for Cloud Microservices Optimization and Performance Analysis of AWS FPGA F1

Shaik Alibasha[1]
Department of Computer Science and Engineering
Koneru Lakshmaiah Education Foundation
alialibasha950@gmail.com

Yaswanthsrinivas Gurram[2]
Department of Computer Science and Engineering
Koneru Lakshmaiah Education Foundation
yaswanthsrinivasg@gmail.com

Kaipakam Gayathri[3]
Department of Computer Science and Engineering
Koneru Lakshmaiah Education Foundation
gayivenky5112001@gmail.com

Nama Deepak Chowdary[4]
Department of Computer Science and Engineering
Koneru Lakshmaiah Education Foundation

Abstract - The cloud's Flexibility and scalability, has significantly altered the architecture, implementation, and processing power of legacy data centers. Microservices are the atom network services that cloud architectures provide. Microservices, in contrast to virtual machines, can be implemented as tiny resource footprint programmer such as container or even lower as unkennels. Field Programmable Arrays (FPGA) boards were first used in servers to effectively offload the execution of computation-intensive applications. FPGAs are increasingly regarded as processing resources that meet cloud standards. FPGAs, however, cannot be used to run several microservices simultaneously in today's cloud data centers. This greatly restricts how effectively microservices can be deployed. We also Used AWS based Architecture to Explore the concept of FPGA in Cloud.

***Key Words*:** *Microservices, AWS, FPGA, Cloud Computing*.

1.INTRODUCTION

Over the past ten years, changes in the technology and architecture of old data centers have given rise to new kinds of cloud systems. S. It was important to create and create new technologies with new functionality, such as management and automation software, in order to achieve this goal. These technologies' key capabilities rely on the system virtualization of the computing units that make up servers in data centers. While virtualization benefits general-purpose apps, it has a negative impact on the performance of the system for time-critical applications [1] [2].

FPGAs are used in Amazon EC2 F1 instances to deliver custom hardware accelerations. F1 instances are simple to programme and include everything you require to create, replicate, troubleshoot, and collate your special hardware code, such as an FPGA Development company AMI and cloud support for hardware level development. Using F1 instances to implement equipment accelerations can help solve complex subject, technology, and enterprise issues that necessitate high bandwidth, augmented networking, and extremely high compute capabilities. Genomics, search/analytics, imagery and video processing, information security, electronic design mechanization (EDA), picture and file compression, and big data analytics are instances of desired applications that can advantage from F1 instance acceleration.

Microservices enable integration and delivery, making it simple to experiment with new ideas and roll it back if something doesn't work. The reduced cost of failure encourages experimentation, makes code updates easier, and reduces time for some new features. By breaking down software into tiny, well-defined modules, teams can use features for multiple purposes. A provider written for one feature can serve as a foundation for another. Devs can develop new capabilities without having to write code from scratch, allowing an implementation to bootstrap itself.

Cloud resource providers such as Amazon had also incorporated Field Programmable Arrays (FPGA) in their data centers to achieve the necessary processing efficiency. FPGAs, like ASICs, can be configured to process specific applications, except that FPGAs can be reprogrammed at runtime to change the functionality of the hardware circuitry. The use of FPGAs in public cloud bridges the divide between the handling flexibility provided by GPPs and the obtain accurate provided by ASICs. FPGAs are used in many today's cloud infrastructures, either as full computational power for the public cloud itself, as AWS does, or as processing resources made available to end users, as almost all other cloud providers do.

The microservices approach simplifies complex software systems by dividing them into subcomponents and disseminating these components across multiple computing servers. An application in this approach is made up of numerous small independent services, each of which runs on its own process. Microservices in cloud infrastructure promote modularity, flexibility, and distributed software components. Indeed, unlike memory or Processor Core Unit (CPU) resources, FPGAs available today in a public cloud are not contemporaneously shared among having to run microservices on a single host. To gain cloud paradigms for accelerated applications, the underlying hardware platform must provide dedicated support and services to share processing elements without compromising performance.

2. RELATED RESEARCH

Today's cloud service providers for information technology (IT) are widely using FPGAs. We could see that when FPGAs are deployed, two main application fields are the focus. The first goal is to speed up infrastructure operation, and the second goal is to provide FPGA as a service to end users for their own acceleration requirements.

2.1 FPGA optimization for use with IT infrastructure

To the greatest of our knowledge, there is only one example of FPGA being used directly for IT infrastructure rather than end-use. Microsoft's hyperscale acceleration fabric is powered by Catapult [9]. The project began in 2010 with the goal of developing a supercomputing substrate capable of providing computing acceleration in a variety of domains such as system, security, cloud services, and artificial intelligence. The Catapult acceleration architecture is distributed because each data center server has one connected FPGA. In parallel, all interconnected FPGAs form an elastic reprogrammable acceleration fabric that enables the use of a single FPGA or up to multitudes of FPGAs for a particular service with 40 Gigaops/W efficiency for implemented at-scale accelerators [3]

3. FPGA FOR CLOUD MICROSERVICES

In this paper, we present a novel approach for bridging the gap between FPGA-based acceleration technology and software microservice architecture running on servers. We have defined a systems design in which microservices can access and start sharing the same FPGA for feature offloading and acceleration. Multiple access and acceleration sharing can occur without service disruption under certain operating conditions, which we will evaluate and define in the following section. The server will process virtual features in the form of containers, while the FPGA panel connected to the server via a PCIe 5.0 interface will offload functions [4].

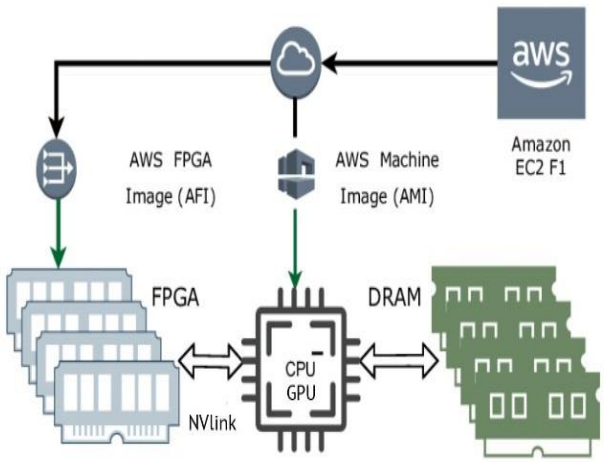

Fig 1: Architecture Diagram of AWS FPGA Instance

The FPGA is divided into many motion slots that are straight and independently available from the containers in order providing flexible and decided to share acceleration resources. To ensure optimal data rates and minimal latency for the expedited distributed applications, data-plane and control-plane communications are routed to the FPGA via different interfaces [6].

4. MICROSERVICES IN APPLICATION DEVELOPMENT.

Microservices are a design approach for developing applications. Microservices are dispersed and loosely coupled as an integrated framework, so changes made by one team will not tear the entire app. The advantage of using microservices is that advancement teams can quickly build new app components to meet changing business requirements[5].

5. RESULT

The Aim of the paper is to also provide the techniques that we are used to build a new application for performance analysis of AWS FPGA.The following evaluation's primary goal is to show how current FPGAs may share logic resources among a number of microservices. Also, this test tries to determine the highest data transfer rate possible for multiple concurrently operating containers between both the server and FPGA.

Throughput results for slots accessed in parallel, The experiment's goal was to estimate the total throughput that could be achieved by running different acceleration slots in parallel. The throughput for up to 7 containerized features trying to run on eight different Dockers is computed.

The experiment's goal was to determine how running up to eight acceleration slots concurrently might affect each slot's throughput. The outcomes are displayed. From the preliminary findings we discussed above, it was clear that one speed slot could handle a large bandwidth of 1,6 GB/s. This was possible because the PCIe 5.0 interface's entire bandwidth was made available to a microservice. Also, we noted that the PCIe 5.0 interface's maximum possible throughput was 8 GB/s. This most recent experiment demonstrates that even though the PCIe 5.0 bandwidth of 8GB/s also isn't reached when running several acceleration slots concurrently, the maximum throughput of 1,6 GB/s per acceleration slot cannot be achieved.

We also Developed a New application based on Microservices with AWS microservices (FPGA enabled) services to do performance analysis. The F1 FPGA done an excellent work than other competitors in the market Ali f1.

6. CONCLUSION

The design of an FPGA-based system that allows for the simultaneous operation of many microservices with connectivity to a common FPGA serving as a local processor is given in this paper. The FPGA has been successfully divided into several momentum spaces that are shared by numerous active server-side microservices. Due to the partial reconfiguration capabilities of the FPGA, the functionality of the slot as implemented in the FPGA hardware can be modified during runtime. To make these capabilities possible, we have created and prepared.

Using high-performance computation on Amazon EC2 F1 instances, as opposed to on-premises systems, gives you nearly infinite ability to scale out your architecture and the freedom to change resources quickly as needed. You may create as many FPGA instances as necessary in minutes, tailor your resources to fit the needs of your program, and only pay for what you actually use.

Aws FPGA F1 instance can speed up a range of compute-bound workloads by up to 100X when compared to CPUs. To accelerate their computing pipelines, clients may quickly and simply identify, test, and deploy custom boosters from the Amazon Marketplace. Coding FPGAs is not required for F1-based solutions developed by F1 technology businesses because they are bundled like any other Ec2 software.

REFERENCES

1. D. Merkel, “Docker: Lightweight Linux containers for consistent development and deployment,” Linux J., vol. 2014, no. 239, Mar. 2014.
2. D. K. Rensin, Kubernetes - Scheduling the Future at Cloud Scale, 1005 Gravenstein Highway North Sebastopol, CA 95472, 2015. [Online].
3. A. M. Caulfield et al., “Configurable clouds,” IEEE Micro, vol. 37, no. 3, pp. 52–61, 2017.
4. Xilinx. (2017) Vivado design suite user guide: Partial reconfiguration. [Online]. Available: https://www.xilinx.com/support/documentation/sw\ manuals/xilinx2017\ 4/ug909-vivado-partial-reconfiguration.pdf
5. Zheng, Ling & Wei, Bo. (2018). Application of microservice architecture in cloud environment project development. MATEC Web of Conferences. 189. 03023. 10.1051/matecconf/201818903023.
6. Choi, J., Lian, R., Li, Z., Canis, A., & Anderson, J. (2018). Accelerating memcached on AWS Cloud FPGAs. *Proceedings of the 9th International Symposium on Highly-Efficient Accelerators and ReconfigurableTechnologies*. https://doi.org/10.1145/3241793.3241795