A Big Data Software Paradigm for Heterogeneous Cloud Deployments

Christos Kotselidis[1], Sotiris Diamantopoulos[2], and Georgios Mylonas[3]

[1] The University of Manchester, United Kingdom
christos.kotselidis@manchester.ac.uk
[2] EXUS Software Limited, United Kingdom
s.diamantopoulos@exus.co.uk
[3] Computer Technology Institute and Press Diophantus, Patras, Greece
mylonasg@cti.gr

Abstract. E2Data is an EU-funded research project aiming to provide a new Big Data software paradigm for achieving maximum resource utilization for heterogeneous cloud deployments without requiring developers to change their code. It essentially addresses the fundamental question of how modern Big Data frameworks can dynamically and transparently exploit heterogeneous hardware accelerators. The E2Data solution takes a cross-layer approach by allowing vertical communication between the four key layers of Big Data deployments: application, Big Data software, scheduler/cloud provider, and execution run time. For this purpose, four different use-case scenarios have been selected to prove the applicability of the project's approach: health analytics, natural language processing, green buildings and biometric security. To achieve those goals, Apache Flink and the Tornado VM are used in the project as the building blocks for enabling heterogeneous execution for Big Data frameworks.

Keywords: Big Data · Heterogeneous Cloud Deployments · Resource Utilization · Apache Flink · TornadoVM.

1 Introduction

Several Big Data frameworks have emerged to address our ever-increasing data generation and processing needs. Regardless of the programming model or features that each framework offers, their scalability is mainly achieved through the following techniques: 1) scale-up by increasing the resources of a single node (e.g., Saber [7]) scale-out by increasing the number of nodes (e.g., Apache Spark [12], Flink [4], Storm [2]), or 3) manual implementation of code optimizations specific to the underlying hardware, like GPU offloading. Typically, scale-up and scale-out approaches concern CPU-only deployments, while manual scalability in the form of ad-hoc optimizations and hardware acceleration targets the, still-emerging, heterogeneous data centers infrastructures. Hardware accelerators are constantly gaining popularity for Machine Learning and Big Data Analytics workloads, since they outperform general-purpose CPU-based implementations, due to their massive capabilities for parallel execution and energy efficiency.

In this context, cloud vendors are offering specialized hardware accelerators along with general purpose resources (e.g., Amazon's EC2 Elastic GPUs or FPGA instances, Google's TPU). Recently, cloud/cluster management software systems such as Apache Yarn [10] and Mesos [1] have provided support for heterogeneous hardware through their API over bare metal, Virtual Machines or even Docker containers. The exploitation of heterogeneous hardware accelerators by Big Data applications is a challenging task. This is, mainly, attributed to: 1) the CPU-only homogeneous design assumptions of Big Data frameworks; 2) the fragmentation of programming models across different devices; and 3) the lack of compiler and run-time support for heterogeneous hardware, by the underlying execution engines of Big Data frameworks - mainly Java Virtual Machines (JVMs).

2 Project Description

E2Data [3] (European Extreme Performing Big Data Stacks) is a H2020 research project running from January 2018 to December 2020. The E2Data consortium consists of nine partners from the UK, Germany, Luxembourg and Greece, bringing into the equation powerful software components, such as Apache Flink and the Tornado Virtual Machine[4] [5], [6]. To enable the exploitation of the heterogeneous hardware accelerators, E2Data proposes novel pluggable extensions to existing software components of a Big Data software stack. That said, the envisioned stack will be capable of adapting itself to the underlying heterogeneous hardware resources while retaining a unified, high-level programming model. Finally, E2Data's extensions are designed to be technology-independent and could potentially be adopted by any technology vendor. E2Data dynamically exploits heterogeneous hardware, such as GPUs and FPGAs, aiming to:

- Enable dynamic heterogeneous compilation of arbitrary code.
- Follow a full-stack vertical approach by enhancing state-of-the-art Big Data frameworks.
- Design a hardware-aware intelligent elastic system [], which makes it possible to: a) Profile results, communicate to scheduler, and assess decisions, b) Fall back and recompile on-the-fly, c) Iterate until the AI enabled scheduler finds the "best" possible execution configuration.

The proposed stack will be evaluated against applications from four different domains: 1) Health Analytics: A real-time streaming use case for predicting patients hospital re-admissions, 2) Natural Language Processing: Sentiment analysis and opinion mining to enable fraud detection, 3) Green Buildings: Enabling energy efficient buildings based on analytics of data derived from Internet of Things deployed sensors, and 4) Biometric Security: Real-time video recognition to enable biometric authentication.

[4] https://github.com/beehive-lab/TornadoVM

Health Analytics: It is necessary to improve the predictive capability of a hospital readmission risk prediction algorithm. To achieve this, the patient discharge profile is enhanced with profiles of highly correlated patients (in terms of recent hospital activity). The patient correlations are established based on their medical conditions. The algorithmic solution enters into time sensitive matrix calculations which need to be accelerated appropriately.

Natural Language Processing: Text processing aims to extract knowledge; reasoning, correlation with event streams, sentiment analysis, data cleansing, interpretation and organization of contents are indicative useful applications, which can be useful in several business domains (finance, tourism, telecommunications, marketing). In many cases, such processing must deliver results under strict time constraints; however, the utilized algorithms are often complex and data volume can be extremely high. In E2Data, we focus on accelerating critical code performing fuzzy matching among words and expressions; these algorithms are falling in the critical path in the knowledge extraction process and, therefore, acceleration is considered as a solution towards enhanced performance.

Green Buildings: The Green Buildings use case provides a scenario for achieving energy efficient buildings based on analytics of real time stream data derived from Internet of Things (IoT) deployed sensors in a number of public schools. In this context, Big Data analytics algorithms and techniques will be deployed in E2Data in order to achieve higher levels of computational efficiency that will enable processing pipelines for the real-time monitoring of the buildings' energy behaviour.

Biometric Security: Biometric authentication, using facial recognition, is rapidly becoming a mainstream method of authenticating customers for high value transactions, such as the creation of Bank Accounts, issuing of Travel Visas and unmanned border crossing by pre-registered users. Such processes are coupled with tight performance requirements and Service Level Agreements (SLAs) to ensure the best possible user experience. E2Data will optimize both the cost base of the platform and automate the performance optimization of code, something that until now has required skilled and expensive developers.

3 Current Status

The project utilizes the following core software components:

- Apache Flink: a batch and stream data processing engine, which is Java-based and scales up/out on CPU clusters and cloud deployments.
- TornadoVM: a heterogeneous programming framework developed at the University of Manchester that enables heterogeneous hardware acceleration of Java on GPUs and FPGAs [9], [11].
- Heterogeneous Aware Intelligent Scheduler (HAIER [8]): a component for providing intelligent scheduling of code on hardware accelerators inside a heterogeneous cloud deployment.

The combination of Apache Flink, TornadoVM, and HAIER will enable heterogeneous execution of Big Data frameworks. At this point in time, there have been two public releases of the Tornado VM since the project began on January 2018 providing support for GPUs (NVIDIA, AMD, INTEL), FPGAs (INTEL) and CPUs (INTEL, AMD). TornadoVM currently achieves in certain workloads an up to 500x speedup on GPUs and 10x on FPGAs, which make a very convincing case for the necessity of tools like the ones produced in E2Data.

4 Conclusions - Future Work

E2Data aims to enhance the current software landscape with respect to the execution of Big Data workloads over heterogeneous cloud deployments, in light of the recent advancements in hardware and the speedup opportunities they provide. Currently available tools in many cases, such as in Java-based implementations, cannot utilize these advancements in an easy manner and this is where E2Data fits into the picture. The project will produce tools to deliver a performance increase in parallelizable code parts on GPUs and FPGA, enabling maximum resource utilization for heterogeneous cloud deployments without affecting current programming norms.

Acknowledgment

This work has been supported by the EU research project "European Extreme Performing Big Data Stacks" (E2Data), funded by the European Commission under H2020 and contract number 780245.

References

1. Apache Mesos and GPUs, https://www.nvidia.com/object/apache-mesos.html (retrieved online, July 2019)
2. Apache Storm, https://storm.apache.org (retrieved online, July 2019)
3. European extreme performing big data stacks project page, https://e2data.eu/ (retrieved online, July 2019)
4. Carbone, P., Katsifodimos, A., Ewen, S., Markl, V., Haridi, S., Tzoumas, K.: Apache flink: Stream and batch processing in a single engine. IEEE Data Eng. Bull. **38**, 28–38 (2015)
5. Clarkson, J., Fumero, J., Papadimitriou, M., Zakkak, F.S., Xekalaki, M., Kotselidis, C., Luján, M.: Exploiting High-performance Heterogeneous Hardware for Java Programs Using Graal. In: Proceedings of the 15th International Conference on Managed Languages & Runtimes (2018)
6. Fumero, J., Papadimitriou, M., Zakkak, F., Xekalaki, M., Clarkson, J., Kotselidis, C.: Dynamic Application Reconfiguration on Heterogeneous Hardware. In: Proceedings of the 15th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (2019)

7. Koliousis, A., Weidlich, M., Castro Fernandez, R., Wolf, A.L., Costa, P., Pietzuch, P.: Saber: Window-based hybrid stream processing for heterogeneous architectures. In: Proceedings of the 2016 International Conference on Management of Data. pp. 555–569. SIGMOD '16, ACM, New York, NY, USA (2016). https://doi.org/10.1145/2882903.2882906, http://doi.acm.org/10.1145/2882903.2882906
8. Mytilinis, I., Bitsakos, C., Doka, K., Konstantinou, I., Koziris, N.: The vision of a heterogenerous scheduler. In: 2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom). pp. 302–307 (Dec 2018). https://doi.org/10.1109/CloudCom2018.2018.00065
9. Papadimitriou, M., Fumero, J., Stratikopoulos, A., Kotselidis, C.: Towards prototyping and acceleration of java programs onto intel fpgas. In: 2019 IEEE 27th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). pp. 310–310 (April 2019). https://doi.org/10.1109/FCCM.2019.00051
10. Vavilapalli, V., Murthy, A., Douglas, C., Agarwal, S., Konar, M., Evans, R., Graves, T., Lowe, J., Shah, H., Seth, S., Saha, B., Curino, C., O'Malley, O., Radia, S., Reed, B., Baldeschwieler, E.: Apache hadoop yarn: Yet another resource negotiator. In: Proceedings of the 4th Annual Symposium on Cloud Computing, SoCC 2013. Association for Computing Machinery (ACM), United States (1 2013). https://doi.org/10.1145/2523616.2523633
11. Xekalaki, M., Fumero, J., Kotselidis, C.: Challenges and proposals for enabling dynamic heterogeneous execution of big data frameworks. In: 2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom). pp. 335–341 (Dec 2018)
12. Zaharia, M., Xin, R.S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S., Franklin, M.J., Ghodsi, A., Gonzalez, J., Shenker, S., Stoica, I.: Apache spark: A unified engine for big data processing. Commun. ACM **59**(11), 56–65 (Oct 2016). https://doi.org/10.1145/2934664, http://doi.acm.org/10.1145/2934664