2021 IEEE 17th International Conference on eScience (eScience) | 978-1-6654-0361-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/eScience51609.2021.00053

Enhancing Automated FaaS with Cost-aware Provisioning of Cloud Resources

Matt Baughman*, Ian Foster*†, Kyle Chard*†
*Department of Computer Science
University of Chicago, Chicago, Illinois
†Data Science and Learning Division
Argonne National Laboratory, Lemont, Illionois

Abstract—**Compute resources are becoming more diverse, more specialized, and more physically distributed every day. To properly use these resources, the new paradigm of serverless computing aims to abstract away the complexity associated with resource configuration and workload deployment. Building on this serverless architecture are efforts for automated resource selection and automated task distribution and coordination. As computing resources advance, so too does application-specific optimizations, as is the case of deep learning on GPUs or even the more specialized TPUs. To better navigate the efficient and effective use of these diverse resources, we present the addition of automated, cost-aware provisioning of cloud resources to a state-of-the-art automated serverless framework. By automating the selection, provisioning, and configuration of cloud resources, our framework, which we call DELTA+, will enable truly cost-aware usage of the cloud by navigating complex computational tradeoffs. In this proposal, we will outline how we plan to introduce burstable cloud computing to automated serverless infrastructure and to present our initial findings with respect to system design and performance.**

1. Introduction

The serverless computing paradigm has been gaining significant ground in recent years and continues to expand in markets like edge computing [1] and burstable computing. With this growth in serverless infrastructure, we also see an increase in the availability and accessibility of the cloud. However, most commercial clouds develop only platform-specifc serverless framworks, making unified usage of multiple platforms difficult. The rise of open-source Function-as-a-Service (FaaS) frameworks allow users to configure and add their own resources to be used in a serverless ecosystem.

Between cloud growth and increasingly diverse end-user hardware, resource selection and configuration has become even more of a complex problem. Additionally, coordination of tasks within a serverless ecosystem at large scales must be automated under a global optimizer as common heuristics become less effective with increased heterogeneity.

1.1. Cloud and Spot Market

In the past three years, Amazon Web Services (AWS) has gone from offering around 100 instance types to offering nearly 400 today [2]. This rapid increase highlights the growing problem of intelligent and efficient resource selection and configuration, particularly for high performance workloads. Additionally, AWS offers terminable "spot" instances [3], which represent AWS's unused capacity that it offers at a reduced cost. These spot instance present a new tradeoff of cheaper per hour costs but significantly reduced reliability.

1.2. funcX and Globus

Given its aim to serve as a federated serverless framework, we have built our automated FaaS platform on the funcX high performance FaaS framework. funcX is designed to function on most compute resources that can run Python and to scale and scale dynamically to accommodate high throughput and high performance serverless workloads.

In addition to funcX, we use Globus to coordinate file and data transfers within DELTA+ ecosystems. Globus enables secure and automated distributed data management [4] and its GlobusAuth [5] is what funcX uses to enable secure access to serverless endpoints [6].

1.3. DELTA+

To enable the automated FaaS, we presented our DELTA (Distributed Execution of Lambdas with Tradeoff Analysis) framework earlier this year [7], shortly followed by DELTA+ [8]. The original DELTA framework aimed to set baseline infrastructure for automated FaaS in modern distributed environments. DELTA demonstrated its capability across edge, HPC, cloud, and commodity resources.

As DELTA only incorporated time as an optimization parameter, we developed DELTA+ which uses configurable, multi-dimensional notions of cost as well as cost and time constraints to better characterize end-user needs [8].

The remainder of this proposal will be organized as follows: Section 2 will demonstrate the importance of adding automated resource provisioning to DELTA+, Section 3 will explain what we have currently done to achieve this goal and what we plan to accomplish for poster presentation, Section 4 gives a brief outline of related works, and Section 5 will summarize our planned contributions.

978-1-6654-0361-0/21/$31.00 ©2021 IEEE
DOI 10.1109/eScience51609.2021.00053

2. Motivation

Our previous work in this area included developing a framework to automate the coordination of tasks within a FaaS ecosystem then extending that framework to include optimization metrics beyond execution durations. In order to realize one of the most important use-case of serverless computing—bursty workloads—we are currently expanding our framework further to enable predictive provisioning of cloud resources for workloads. In other words, the heterogeneous and dynamic nature of clouds and cloud usage make it infeasible or impractical to have reserved compute resources sitting idle.

Additionally, our previous work regarding resource selection, workload profiling, and instance price prediction leads naturally to these works' integrations with our existing DELTA+ framework. By incorporating automated resource provisioning, configuration, and release, we further abstract the need for end-users to navigate the complexities of cloud resource management. Finally, the addition of using AWS spot instances will enable further reduction in costs and increase in user-accessible compute.

3. Expected Results

To demonstrate our design, we have exposed funcX's built-in AWS resource provider functionality to DELTA+ and associated the necessary cost and time-based elements with the framework. We must now integrate the predictive models developed for resource selection in past projects. The information provided by these models will enable DELTA+ to properly consider the use of dynamically provisioned EC2 on-demand and spot instances.

To evaluate the system, we will use and expand the benchmarks from the original DELTA paper [7]. We anticipate the data we collect from these tests will be adequate to demonstrate the superiority of performance and cost efficiency of DELTA+ over previous iterations of itself. Additionally, we will use this data to further inform our research into building out more developed predictive models for robust resource selection and provisioning.

4. Related Works

Regarding the use of automated resource provisioning, most popular FaaS platforms, like AWS Lambda [9], abstract away user need to configure endpoints and deploy workloads. Similarly, the opensource FaaS framework DELTA+ is built on—funcX—uses the Parsl parallel scripting library to allow for dynamic scaling of many HPC and commercial cloud resources.

Beyond serverless, burstability in cloud resources is found most commonly in AWS's *burstable performance instances* that allow for increases in computational power for some restricted duration of time in a given period. There has been additional work in this space around a more generalized model of computational profiling, prediction, and provisioning in the SCRIMP framework [10], which was designed to enable full-cycle cloud usage in a cost-aware method, though is more aimed at high performance workloads. Finally, the Seagull system [11] incorporates many cost-aware elements with a focus on QoS for enterprise.

5. Conclusion

This poster will represent an overview of DELTA+'s improved design, incorporating dynamic resource provisioning and configuration. The findings from our experimental benchmarks will inform DELTA+'s future development and motivate potential research directions. By presenting this poster in an open forum, we will receive feedback from the community regarding the most impactful directions towards which we can focus our build-up and build-out of DELTA+.

Acknowledgments

This research was supported in part by DOE contract DE-AC02-06CH11357 and by NSF grants 1816611, 2004894, and 1550588.

References

[1] "Amazon Greengrass," aws.amazon.com/greengrass/.

[2] "Amazon Web Services EC2," https://aws.amazon.com/ec2/.

[3] "Amazon Spot Instances," https://aws.amazon.com/ec2/spot/.

[4] B. Allen, J. Bresnahan, L. Childers, I. Foster, G. Kandaswamy, R. Kettimuthu, J. Kordas, M. Link, S. Martin, K. Pickett *et al.*, "Software as a service for data scientists," *CACM*, vol. 55, no. 2, pp. 81–88, 2012.

[5] S. Tuecke, R. Ananthakrishnan, K. Chard, M. Lidman, B. McCollam, S. Rosen, and I. Foster, "Globus Auth: A research identity and access management platform," in *12th Intl Conference on e-Science*, 2016.

[6] R. Chard, Y. Babuji, Z. Li, T. Skluzacek, A. Woodard, B. Blaiszik, I. Foster, and K. Chard, "funcX: A federated function serving fabric for science," in *High-Performance Parallel and Distributed Computing 2020*, 2020.

[7] R. Kumar, M. Baughman, R. Chard, Z. Li, Y. Babuji, I. Foster, and K. Chard, "Coding the computing continuum: Fluid function execution in heterogeneous computing environments," in *Heterogeneity in Computing Workshop 2021*, 2021.

[8] M. Baughman, R. Kumar, I. Foster, and K. Chard, "Expanding cost-aware function execution with multidimensional notions of cost," in *Proceedings of the 1st Workshop on High Performance Serverless Computing*, 2021, pp. 9–12.

[9] "Amazon Lambda," aws.amazon.com/lambda/.

[10] R. Chard, K. Chard, R. Wolski, R. Madduri, B. Ng, K. Bubendorfer, and I. Foster, "Cost-aware cloud profiling, prediction, and provisioning as a service," *IEEE Cloud Computing*, vol. 4, no. 4, pp. 48–59, 2017.

[11] T. Guo, U. Sharma, P. Shenoy, T. Wood, and S. Sahu, "Cost-aware cloud bursting for enterprise applications," *ACM Transactions on Internet Technology (TOIT)*, vol. 13, no. 3, pp. 1–24, 2014.