Scale-up or scale-out: Is there one solution that fits all problem sizes?

Marcel Stolin
marcelstolin@gmail.com

Abstract

Since the beginning of Big Data, batch processing was the most popular choice for processing large amounts of generated data. These existing processing technologies are not suitable to process the large amount of data we face today. Research works developed a variety of technologies that focus on stream processing. Stream processing technologies bring significant performance improvements and new opportunities to handle Big Data. In this paper, we discuss the differences of batch and stream processing and we explore existing batch and stream processing technologies. We also explain the new possibilities that stream processing make possible.

1 Introduction

With the increase of interest in Artificial Intelligence and the growing complexity of modern computing architectures, more powerful resources are needed to adapt to business needs. If an IT environment reaches the limit of its computational power and can‘t serve incoming requests efficiently, it has reached the limit of its scalability [4]. The limit of scalability can be extended by providing more resources. Although, increasing the resources also increases the cost of running QUELLE. The two primary approaches of scaling computing environments are horizontal and vertical scaling. Increasing the computing capacity in a horizontal-elastic environment is called scaling-out, whereas in a vertical-elastic environment it is called scale-up [2]. Both approaches are not exclusive and a computing environment can be designed to scale vertically, horizontally or both [4]. Although both strategies can be used to increase the computation capacity of a computing environment, each strategy follows a different approach.

The contribution of this essay is as follows. First, the two scaling strategies are introduced while mentioning examples. Second, the differences between both scaling strategies and their typical use-cases are described. Third, a summary of this essay is given with a logical conclusion.

2 State of the art

There are many research studies around scalability of IT environments and different use-cases for horizontal and vertical scaling. Rossi et al. [3] created a solution to control the horizontal and vertical elasticity of container-based applications using Reinforcement learning WHATS THE RESULT.

The cloud computing architecture has contributed to the scalability of modern computing environments. Cloud computing makes it easy to increase the computing capacity of a computing environment on demand [5].

3 Background

This section provides information about the scale-up and scale-out approaches.

3.1 Scale-up

Scale-up refers to improving the hardware of individual nodes [4]. By adding more powerful resources to a node, a node can take more throughput and perform more specialized tasks [2]. Increase the computing capacity of a node can include adding memory or adding more CPU cores [4]. Due to the low complexity of scaling-up, it is the mos applied scaling approach [4].

3.2 Scale-out

Scale-out refers to horizontal scaling by cloning instances of a service or data in an IT environment. By cloning instances, the work can be distributed across instances to handle an increasing number of requests. A new clone is able to perform the same work and will respond as other clones. In cloud computing, instances are typically deployed using container technologies. Cloning containers is easy due to their lightweight architecture [2].

A typical example for using a scale-out strategy is, to create new instances of a database to serve read-operations for rapidly increasing requests.

4 Scale-up or scale-out

This section describes a typical use case where scaling is necessary. It gives an overview about how both scaling approaches to solve the problem and concludes with the most logical solution.

If a computing environment reaches the limitations of its hardware capacity, options are to scale-up the environment by adding more powerful resources or scale-out to distribute the work across multiple instances. If a computing environment is not designed to scale-out, the only option remaining is to add more powerful hardware [1]. Scaling-up by adding more powerful hardware to the computing environment, is limited by maximum hardware capacity and causes downtimes due to IT resource replacements [5]. An economical threshold can be reached, where buying more powerful hardware is not affordable. Another limitation can be that no more powerful hardware is available by the provider [1, 4]. If the environment is designed to scale-out, buying more powerful resources is not needed. The computing capacity can be increased by adding more nodes which are instantly available. In this case, the scaling-out strategy gives an IT environment the flexibility to scale without increasing the cost [1].

Designing an environment to scale-out, shifts the focus from infrastructure to development. The goal of scaling-out is to increase the computing capacity by combining the computing power of several nodes, which makes a scale-out design more complex compared to a scale-up architecture [4]. The computing capacity of a scale-out architecture is limited by the computing power of each additional node. Therefore, scaling-out is more efficient with homogeneous nodes, where each node adds the same amount of computing power to the system [4]. With non-homogeneous nodes, the complexity of capacity planning and distributing work across nodes grows.

In a cloud environment, scaling-up is less considered by architects due to its significant drawbacks. In general, IT environments should be designed to scale-out in the first place. While the cost of buying more powerful hardware grows proportionally, the cost of investing engineering power to design an environment to scale-out grows linear [1].

5 Conclusion

The ability to scale is a critical characteristic of modern and complex computing environments. With the ability to scale, a computing environment is able to adapt to business needs. Scaling-up and scaling-out are two scaling approaches to increase the computing capacity of an IT environment. Scaling-up is a simpler approach, but it is limited and comes with significant disadvantages according to cost and flexibility. On the other hand, scaling-out adds more complexity to the environment but is able to scale the environment without increasing the cost proportionally. A modern IT environment should be designed to scale-out in the first place to handle performance spikes and keep the cost low.

ES SOLLTE NOCH WAS ÜBER: WENN LIMIT ERREICHT MIT HORIZONTAL -¿ VERTICAL IST IMMERNOCH NE MÖGLICHKEIT

References

[1] Abbott Martin L. and Fisher Michael T. *Scalability Rules: 50 Principles for Scaling Web Sites*. Addison-Wesley Professional, May 2011.

[2] Abbott Martin L. and Fisher Michael T. *The Art of Scalability: Scalable Web Architecture, Processes, and Organizations for the Modern Enterprise, Second Edition*. Addison-Wesley Professional, June 2015.

[3] F. Rossi, M. Nardelli, and V. Cardellini. Horizontal and vertical scaling of container-based applications using reinforcement learning. In *2019 IEEE 12th International Conference on Cloud Computing (CLOUD)*, pages 329–338, 2019.

[4] Bill Wilder. *Cloud Architecture Patterns*. O'Reilly Media, Inc., September 2012.

[5] Mahmood Zaigham, Puttini Ricardo, and Erl Thomas. *Cloud Computing: Concepts, Technology & Architecture*. Pearson, May 2013.