World Academy of Science, Engineering and Technology
International Journal of Computer and Systems Engineering
Vol:17, No:03, 2023

Modelling of Reactive Methodologies in Auto-Scaling Time-Sensitive Services With a MAPE-K Architecture

Authors : Óscar Muñoz Garrigós, José Manuel Bernabeu Aubán

Abstract : Time-sensitive services are the base of the cloud services industry. Keeping low service saturation is essential for controlling response time. All auto-scalable services make use of reactive auto-scaling. However, reactive auto-scaling has few in-depth studies. This presentation shows a model for reactive auto-scaling methodologies with a MAPE-k architecture. Queuing theory can compute different properties of static services but lacks some parameters related to the transition between models. Our model uses queuing theory parameters to relate the transition between models. It associates MAPE-k related times, the sampling frequency, the cooldown period, the number of requests that an instance can handle per unit of time, the number of incoming requests at a time instant, and a function that describes the acceleration in the service's ability to handle more requests. This model is later used as a solution to horizontally auto-scale time-sensitive services composed of microservices, reevaluating the model's parameters periodically to allocate resources. The solution requires limiting the acceleration of the growth in the number of incoming requests to keep a constrained response time. Business benefits determine such limits. The solution can add a dynamic number of instances and remains valid under different system sizes. The study includes performance recommendations to improve results according to the incoming load shape and business benefits. The exposed methodology is tested in a simulation. The simulator contains a load generator and a service composed of two microservices, where the frontend microservice depends on a backend microservice with a 1:1 request relation ratio. A common request takes 2.3 seconds to be computed by the service and is discarded if it takes more than 7 seconds. Both microservices contain a load balancer that assigns requests to the less loaded instance and preemptively discards requests if they are not finished in time to prevent resource saturation. When load decreases, instances with lower load are kept in the backlog where no more requests are assigned. If the load grows and an instance in the backlog is required, it returns to the running state, but if it finishes the computation of all requests and is no longer required, it is permanently deallocated. A few load patterns are required to represent the worst-case scenario for reactive systems: the following scenarios test response times, resource consumption and business costs. The first scenario is a burst-load scenario. All methodologies will discard requests if the rapidness of the burst is high enough. This scenario focuses on the number of discarded requests and the variance of the response time. The second scenario contains sudden load drops followed by bursts to observe how the methodology behaves when releasing resources that are lately required. The third scenario contains diverse growth accelerations in the number of incoming requests to observe how approaches that add a different number of instances can handle the load with less business cost. The exposed methodology is compared against a multiple threshold CPU methodology allocating/deallocating 10 or 20 instances, outperforming the competitor in all studied metrics.

Keywords : reactive auto-scaling, auto-scaling, microservices, cloud computing
Conference Title : ICDSC 2023 : International Conference on Distributed and Scalable Computing
Conference Location : Barcelona, Spain
Conference Dates : March 06-07, 2023