

# 쿠버네티스 다중 클러스터 환경에서의 컴퓨팅 자원 기반 스케줄링 최적화

이인기, 김태훈, 김영한\*

\*송실대학교

ilsf1004@dcn.ssu.ac.kr, bmxm1123@dcn.ssu.ac.kr, \*younghak@ssu.ac.kr

## Computing Resource Based Scheduling Optimization in Kubernetes Multi-Cluster Environment

Lee In Gi, Kim Tae Hoon, Kim Young Han\*

\*Soongsil Univ.

### 요 약

쿠버네티스가 컨테이너 기반 클라우드의 일반적인 오케스트레이션 플랫폼이 되면서 쿠버네티스의 컴퓨팅 구성 기반인 클러스터가 더욱 많은 응용프로그램의 기반환경이 되고 있다. 또한, 고가용성을 제공하기 위해 클러스터를 다중으로 구성하여 사용하고자 하는 수요 또한 증가하고 있다. 기존의 단일 클러스터 환경과는 달리 다중 클러스터 환경은 분산된 인프라 환경에 대규모 워크로드를 효율적으로 배포하기 위해 클러스터 차원의 스케줄링을 고려해야 한다. 이에 본 논문에서는 다중 클러스터 환경에서 단일 API 를 통해 분산형 인프라의 전체적인 컴퓨팅 자원에 기반한 스케줄링 최적화를 위한 구조를 제안한다.

### I. 서 론

최근 컨테이너화된 애플리케이션의 수요가 증가함에 따라 높은 수준의 가용성을 제공하기 위해 쿠버네티스 클러스터를 다중으로 구성하여 관리할 수 있다. 다중 클러스터에 의해 애플리케이션 및 워크로드에 대한 확장성 및 가용성은 증가했지만 클러스터의 수가 증가하면서 각 클러스터를 최신 상태로 유지하고 관리하기 위해 고려해야 할 구성요소도 늘어나게 되었다[1]. 이처럼 다중 클러스터 환경에서는 단일 클러스터 환경과는 달리 여러 구성요소를 고려한 클러스터 차원에서의 최적화된 스케줄링을 제공해야 한다. 다중 클러스터로 구성된 분산형 인프라의 경우, 컴퓨팅 자원을 낭비하지 않으면서 애플리케이션이 실행될 최상의 조건을 가진 노드에 워크로드를 배포하기 위해 워크로드 배포 시 분산된 인프라의 전체적인 컴퓨팅 자원 상황을 모니터링 해야 한다. 하지만 대규모 워크로드 배포 시 배포가 진행될 때마다 반복적으로 전체 인프라를 모니터링하고, 적합한 노드를 직접 선택하는 과정은 효율적이지 않다. 이를 위해서는 단일 API 로 분산형 인프라의 전체적인 상황을 신속하게 파악하고, 모니터링을 통해 얻은 정보를 기반으로 적절한 스케줄링이 수행되어야 한다.

따라서 본 논문에서는 쿠버네티스 다중 클러스터 환경에서 단일 API 로 분산형 인프라의 전체적인 컴퓨팅 자원을 모니터링하고, 해당 정보를 스케줄링 알고리즘에서 사용할 수 있도록 함으로써 워크로드 배포 시 컴퓨팅 자원 점유 최적화와 스케줄러의 부하 개선을 위한 구조를 설계하고자 한다.

### II. 멀티 클러스터 오케스트레이션

쿠버네티스 다중 클러스터 환경에서 애플리케이션 배포 시 최적의 컴퓨팅 자원 환경을 제공하기 위해서는 (1) 분산형 인프라의 전체적인 컴퓨팅 자원 상황 모니터링 기능, (2) 단일 API 를 통한 중앙 집중식 관리 기능, (3) 각 클러스터 및 노드들의 컴퓨팅 자원 상황에 맞게 애플리케이션을 배포하기 위한 고급 스케줄링 기능을 제공하는 멀티 클러스터 오케스트레이션 솔루션이 필요하다. EMCO 와 Karmada 는 해당 기능들을 제공하는 대표적인 오픈소스 기반 멀티 클러스터 오케스트레이션 솔루션으로, 다중 클라우드 및 쿠버네티스 멀티 클러스터 오케스트레이션 솔루션을 제공한다는 점에서 목적이 같지만 워크로드 배포 시 서로 다른 구조에 의해 클러스터 스케줄링을 구현하는 방식에 차이를 보인다.

#### 1. EMCO

EMCO(Edge Multi-Cluster Orchestrator)는 쿠버네티스용 다중 클러스터 차원에서의 컨트롤 플레인 및 애플리케이션 오케스트레이터로, 여러 클라우드 서비스 제공업체 및 여러 엣지 위치에 걸쳐 쿠버네티스 클러스터들에 대해 클라우드-네이티브 애플리케이션의 의도 기반(intent-based) 배포를 위한 프레임워크를 제공한다. 또한, EMCO 는 워크로드 배포를 위한 클러스터 선택 기능, 모니터링 기능, 네트워킹 및 서비스 메시와 애플리케이션 보안을 위한 인프라 구성 자동화 기능 등 자체 컨트롤러들을 통해 의도 기반 배포를 위한 다양한 기능들을 제공한다[2]. 하지만 이러한 기능들은 EMCO 혹은 사용자가 자체 개발한 컨트롤러들에 의해 수행되기 때문에 기존 스케줄링 알고리즘을 수정하고, 새로운 기능을 추가하기 위해서는 직접 개발하거나 릴리즈 업데이트 시 새로운 기능을 위한 컨트롤러가 제공되어야 한다.

