

User Scenario 4: Kubernetes

[Return to User Experiences](#)

```
* **<color #FF0000><todo @char>Please Review</todo></color>**  
* **<color #FF0000><todo @DDSFmember>Please Review</todo></color>**
```

Overview

This topic is a comprehensive performance evaluation of different Kubernetes CNI plugins for Edge-Based and Containerized Publish/Subscribe applications. First we have preliminary of Kubernetes and its Networking Model. Then I will use a real world example to illustrate the feasibility & challenges of deploying containerized DDS application in a Kubernetes Cluster. We then did systematic experiments to learn the DDS performance in such kind of deployments. Based on the observations we can get some conclusions and insights.

Quick Kubernetes Overview

To fully understand the Kubernetes based service its always good practice to have a brief overview of Kubernetes architecture. A Kubernetes Cluster is composed of a Control Plane/Master Node that leads to at least one worker node. A Node of a Kubernetes Cluster could be a virtual machine or physical machine. A Container is nothing but a process that isolates SIG group and namespace from other processes running on the host. However Kubernetes builds another layer of encapsulation on top of containers called a Pod, or a group of 1 or more containers. From the perspective of networking namespace, each node in the Kubernetes Cluster has an individual seeder and any Pod running on the node has a global unique IP. All containers in the Pod share the same network namespace.

Motivation for the Project

Benefits of Deploying DDS Applications In a Kubernetes Cluster

People really deploy DDS apps in Virtual Machines, but actually there are several benefits for deploying on Kubernetes.

- Container Enables lightweight encapsulation of functional modules, such as DDS' publisher/subscriber capabilities
- Offers resource isolation thereby allowing distributed applications and services to be easily deployed and operated across heterogeneous platforms through sharing standardized container images.
- With Kubernetes, it is easy to deploy, update, scale, and self-heal distributed applications.

Last update: 2021/07/12 15:28 ddsf:public:guidebook:03_user:13_kubernetes https://www.omgwiki.org/dds/doku.php?id=ddsf:public:guidebook:03_user:13_kubernetes&rev=1626118081

From: <https://www.omgwiki.org/dds/> - **DDS Foundation Wiki**

Permanent link: https://www.omgwiki.org/dds/doku.php?id=ddsf:public:guidebook:03_user:13_kubernetes&rev=1626118081

Last update: **2021/07/12 15:28**

