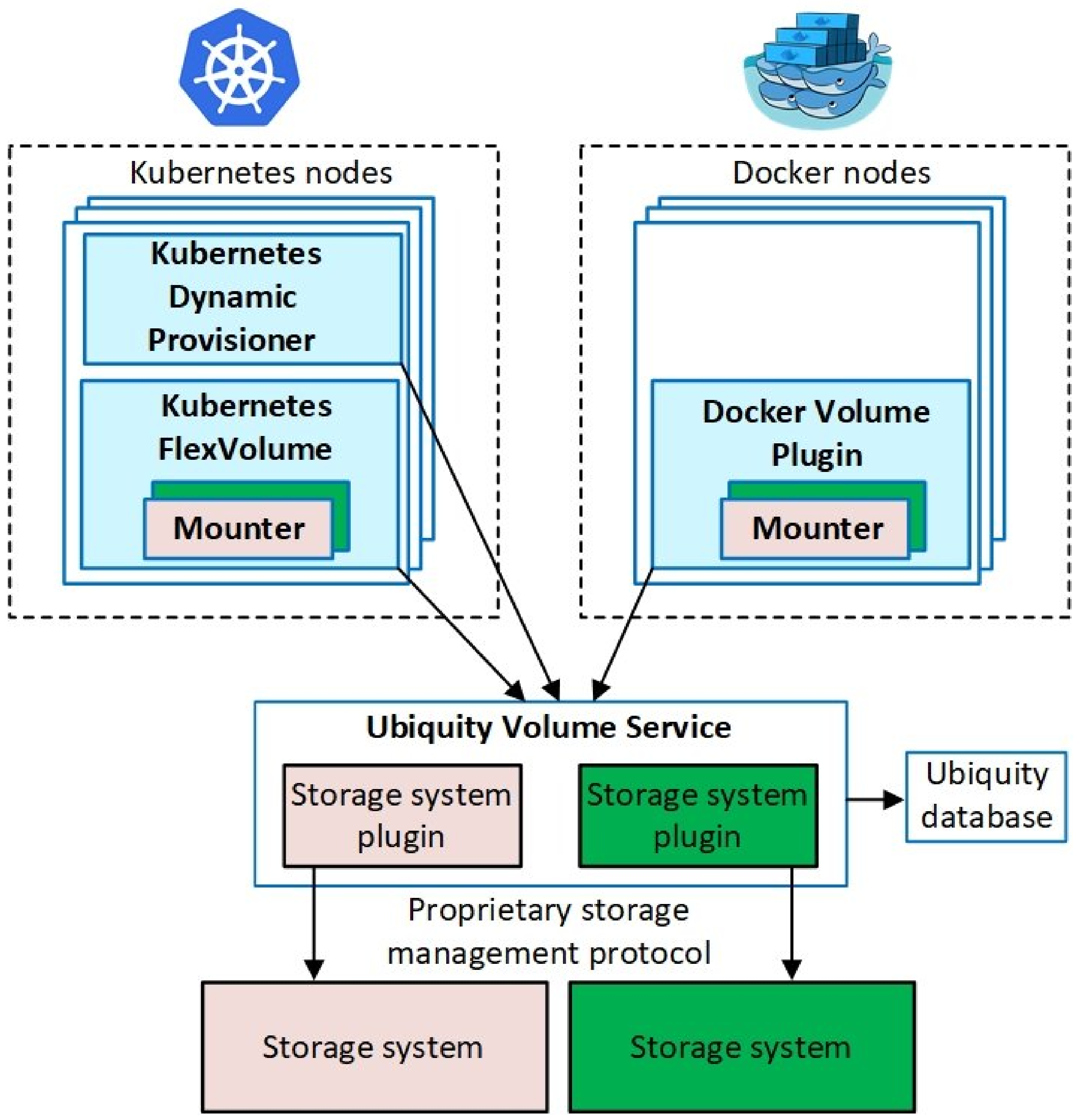
Making container technology work for persistent microservices

December 22 2017, by Heiko Ludwig



The Ubiquity architecture. Credit: IBM

In the technology realm, containers are an approach to running a compute environment, a bit like virtual machines (VM). Users typically prefer containers because they use resources more efficiently, start and stop much faster, and are less expensive to maintain. IBM Research co-developed Ubiquity with the IBM Cloud Storage Solutions team with the objective of providing a container-persistence service and a code base to develop enterprise and provider-grade storage plugins for different block or file storage systems quickly and inexpensively for the most common container orchestrators.

Companies have been using containers primarily to run microservices and applications that do not directly store data. Despite the inherent benefits, containers have one drawback: Unlike a VM they cannot attach or mount persistent, enterprise storage by themselves in production environments.

This problem is exacerbated by container orchestrators, such as Kubernetes, that can create and "kill" containers at a very fast rate in large data centers to dynamically manage compute workloads, e.g., for machine learning or microservices. Container orchestrators have enjoyed rapid adoption in enterprises, both in-house and in the cloud. They help realize the cost and elasticity benefits of container workloads at large scale. Ubiquity is an open-source framework for enabling containers to provide persistent microservices efficiently and at scale for different container orchestrators.

Ubiquity helps to move persistent microservices to containers and thereby realize the same cost and elasticity benefits that other microservices already enjoy. It saves operating costs for these services and makes them respond better to market demand. It is ideal because it provides properties that an enterprise needs such as security, availability management, and the integration with storage management in a framework. This makes Ubiquity quite secure and stable for our enterprise customers, but also gives the open-source community the opportunity to develop enterprise-grade container support for another storage product such as an open-source file system without much effort.

Machine learning systems are an example of an application that can benefit from this type of system. While machine learning frameworks, such as Tensorflow, have become faster using GPUs to process data—and are often deployed in containers—the full processor speed often cannot be utilized because data has to be retrieved from a data service, which can result in processing delays. Mounting storage with training data directly to containers can help with parallel machine learning processes, avoiding idle time of expensive GPUs.

Recently, IBM announced a new product for container orchestrators, that facilitates access to persistent storage, including all IBM block storage products, from container orchestrators such as Kubernetes. It allows block storage to be created and attached dynamically to containers wherever they are placed in a container cluster, combining the advantages of high-end storage products regarding performance and manageability with the dynamics of container orchestrators.

The open-source Ubiquity framework can be downloaded and used directly from Github (GitHub.com/IBM/ubiquity). The IBM Research – Almaden team, including Mohamed Mohamed, Amit Warke, and Robert Engel, encourage extensions to other storage systems and orchestrator frameworks as well as any feedback through Github.

Provided by IBM