Getting started with Milvus - the world's most popular open-source vector database.

@Milvus.io

7/4/2022

Getting Started with Vector Database - Milvus Quickstart

Hey there - welcome back to Milvus codelabs. In the previous tutorial, we provided a brief introduction to Milvus, Milvus' history, as well as the primary differences between Milvus 1.x and Milvus 2.x. We also took a quick tour of the architecture of Milvus 2.x and helped shine some light on how Milvus' architecture allows it to implement all of the required features of vector databases.

Let's get started

If you haven't read the previous tutorials (unstructured data, vector databases, Milvus introduction), I recommend you go ahead and read them. If you have, great. Let's get started with Milvus!

We offer two different modes of deployment: standalone and cluster. In Milvus standalone, all nodes - coordinators, worker nodes, and forward-facing proxies - are deployed as a single instance. For persistent data and metadata, Milvus standalone relies on `MinIO` and `etcd`, respectively. In future releases, we hope to eliminate these two third-party dependencies, allowing everything to run in a single process and removing the need to install third-party dependencies.

Milvus cluster is our full-fledged version of Milvus, complete with separate instances/pods for all eight microservice components along with three third-party dependencies: `MinIO`, `etcd`, and `Pulsar` (Pulsar serves as the log broker and provides log pub/sub services). If you haven't gotten the chance to take a look at the Milvus overview from the previous slide, please do so! It'll help clarify what each of these third party dependencies is used for and why we've included them in Milvus cluster.