

Batch Job

The *Batch Job* pattern is suited for managing isolated atomic units of work. It is based on Job abstraction, which runs short-lived Pods reliably until completion on a distributed environment.

Problem

The main primitive in Kubernetes for managing and running containers is the Pod. There are different ways of creating Pods with varying characteristics:

Bare Pod

It is possible to create a Pod manually to run containers. However, when the node such a Pod is running on fails, the Pod is not restarted. Running Pods this way is discouraged except for development or testing purposes. This mechanism is also known under the names of *unmanaged* or *naked Pods*.

ReplicaSet

This controller is used for creating and managing the lifecycle of Pods expected to run continuously (e.g., to run a web server container). It maintains a stable set of replica Pods running at any given time and guarantees the availability of a specified number of identical Pods.

DaemonSet

A controller for running a single Pod on every node. Typically used for managing platform capabilities such as monitoring, log aggregation, storage containers, and others. See [Chapter 9, *Daemon Service*](#) for a detailed discussion on DaemonSets.

A common aspect of these Pods is the fact that they represent long-running processes that are not meant to stop after some time. However, in some cases there is a need to