

# Introduzione

Kubernetes non è solo un orchestratore. Kubernetes è una rivoluzione vera e propria nel mondo enterprise, che è permeata negli ultimi 10 anni del settore informatico in Italia attraverso un diverso approccio alle infrastrutture, al design, ma anche allo sviluppo delle applicazioni. Questa tecnologia non è soltanto lo strumento di punta di tantissime realtà grandi e piccole, ma è anche il collante di diverse community: come mi piace ricordare, dietro a ognuno di questi progetti c'è un team fatto di persone che crede fortissimamente nel proprio lavoro e che riesce, in qualche modo, a trascinare tantissime persone nella sua sperimentazione e nel suo utilizzo, fino alla completa e devota adozione. Proprio come altre tecnologie figlie di quest'epoca, Kubernetes nasce in un "piccolo" laboratorio ed esplose grazie a quelle persone che, duramente, lavorano anni per renderla un must: non a caso, dietro a molti degli strumenti odierni che sono disponibili tramite dei provider cloud c'è sempre questo strumento. Si tratta, però, di un oggetto piuttosto complesso, che richiede una conoscenza a 360 gradi di ciò che significa poter mettere un'applicazione o un servizio a disposizione di un utente finale: quando inizieremo a interagire con i primi concetti, potremo subito provare la sensazione di avere a che fare con una risorsa che non è solo per persone che sviluppano, o per persone che si occupano di amministrazione di rete, ma è molto di più. Kubernetes *guida* tutto il flusso di lavoro di un'applicazione dal suo sviluppo nel portatile personale, fino a rendere fruibile il sito Web che chiunque può raggiungere tramite il proprio browser.

In questo contesto nasce questo manuale, che vuole essere un percorso di introduzione a questa tecnologia, ma anche un modo per mettersi alla prova con dei casi d'uso reali: grazie all'esperienza maturata vestendo ognuno dei cappelli necessari alla comprensione di questa tecnologia, è stato possibile creare capitoli *ad hoc* per trattare i tanti volti dietro a Kubernetes. Un esempio è il capitolo dedicato alla sua architettura: per chi normalmente si occupa di sviluppo software, il modo in cui l'infrastruttura è stata predisposta e la configurazione su cui installeremo potrebbero essere sconosciuti e anche di scarso interesse. In realtà, per poterci astrarre dal concetto di "applicazione", dovremo aprire il cofano di questo strumento e dare un'occhiata al suo motore: ogni sezione successiva e ogni esempio sono stati curati per rendere la sua comprensione più semplice possibile, soprattutto per tutte le persone che, come una giovane me, non hanno un contesto così forte su una serie di argomenti. Questo manuale non è pensato, però, solo per chi sviluppa, ma anche per chi "costruisce": progettare un'applicazione è un'arte che parte dai dati (o da un'idea), e arriva sempre molto velocemente negli ambienti di produzione,

quindi di tempo per testarla non se ne ha mai abbastanza. Avere a disposizione una serie di casi d'uso reali, insieme a *best practices* che ci permettano di affinare le tecniche che utilizziamo, ci renderà il lavoro sempre più agevole. Questo manuale, infatti, vuole essere un timone sicuro per il vostro viaggio all'interno di Kubernetes: che possa guidarvi con il vento in poppa, ma anche durante la tempesta e nella marea, facendovi sentire sempre forti delle vostre competenze.

## Struttura

Ecco perché questo volume è stato così pensato: dopo una breve introduzione sulla nascita di Kubernetes e sul perché questo strumento abbia fatto da pilota per una serie di progetti successivi (Capitolo 1 e 2), vediamo con gli occhi di un *architect* quali sono i suoi “mattoncini” di base e come installare un ambiente che ci permetta di sporcarci le mani (Capitolo 3 e 4). Cambiamo cappello, e torniamo nel mondo dello sviluppo, analizzando quali sono le risorse che abbiamo a disposizione per poter avviare un'applicazione (Capitolo 5), parametrizzare la sua configurazione (Capitolo 6) e anche renderla disponibile per chi al cluster non sempre vi accede direttamente (Capitolo 7). All'interno del dominio Kubernetes avremo sicuramente bisogno della persistenza, attraverso volumi (Capitolo 8) o attraverso risorse con funzionalità specifiche (Capitolo 9). Torniamo poi nei panni di *architect* ed esploriamo il mondo degli utenti e dei ruoli che ci permettono di rendere più sicuro il nostro cluster (Capitolo 10), anche attraverso strumenti utili a “isolare” gli ambienti e le risorse, nonché a strumenti che ci permettono di adattare il cluster al carico di lavoro e agli imprevisti (Capitolo 14). Kubernetes è stato il motore di una serie di progetti strettamente dipendenti, come Helm e Kustomize, che a oggi sono integrati completamente in questa tecnologia e che vale la pena analizzare con casi d'uso reali (Capitolo 11); questo strumento lascia che le persone che lo utilizzano estendano le sue funzionalità di base con risorse create *ad hoc* dai suoi utenti, che poi possono essere distribuite liberamente o non (Capitolo 12). In ultimo, ma non per importanza, il Capitolo 13 è dedicato all'utilizzo di Kubernetes nel contesto cloud: per avere un occhio sempre verso il futuro, analizziamo quali sono le diverse opzioni presenti tramite i maggiori *vendor*, insieme a una “chicca” tutta italiana.

Le appendici, infine, sono studiate per essere uno strumento utile a completare la visione di insieme sulla community e gli eventi legati al settore, sulle certificazioni, ma anche per avere a disposizione un prontuario di comandi che può tornare utile in un momento di difficoltà.

## A chi è rivolto questo libro

Come anticipato, questo manuale è dedicato a chi vuole avvicinarsi a Kubernetes e lavora nello sviluppo del software, a chi si è sempre occupato di compiti associati alla professione di “sistemista”, ma anche a chi viene dal mondo di Docker e vuole capire come questa tecnologia ne inglobi le infinite potenzialità e le porti in mari ancor più sicuri. Tutti gli esempi che vengono fatti sono dettagliati e sempre indipendenti dalla singola tecnologia: se presente un esempio per un'applicazione scritta in Node.js, le

stesse considerazioni varranno anche se questa utilizza un linguaggio di programmazione diverso, o anche una porta differente.

Inoltre, tutto il codice presente è riportato sul repository relativo a questo manuale e disponibile sul sito <https://github.com/serenasensini/Kubernetes>, di modo che sia più semplice riuscire a rispettare la formattazione degli esempi o che sia anche possibile recuperare parte del codice che, per brevità, viene riportato in maniera parziale. Potrebbe darsi che alcuni degli esempi riportino un'indentazione non precisa: fare affidamento al codice presente online!

## Terminologia

Così come avviene di frequente in questo settore, la maggior parte dei termini con cui avremo a che fare sono in lingua inglese e non facilmente traducibili. In ogni caso, ove possibile, si è preferito l'uso della lingua italiana anche in caso di termini inglesizzati, per favorire chi non vi sia avvezzo: un esempio è la parola “rilascio” al posto di *deployment*, o anche il termine “programmazione” al posto di *schedule*. Il motivo di questa scelta è sì riconoscere l'utilizzo ormai dominante della forma inglese in questo contesto, ma anche far sì che la lettura sia comprensibile a chi è alle prime armi, con la possibilità di accedere sempre alla documentazione relativa a Kubernetes cercando maggiori informazioni tramite i consueti motori di ricerca o nei siti dedicati.

## Errori e feedback

Potete segnalare errori, frasi non chiare o altri riferimenti scrivendo nella sezione *Issues* del repository pubblico indicato in precedenza. Per altre informazioni, è possibile contattare direttamente l'indirizzo [info@theredcode.it](mailto:info@theredcode.it).