

# Docker

Das Praxisbuch für Entwickler und  
DevOps-Teams

» Hier geht's  
direkt  
zum Buch

# DAS VORWORT

# Vorwort

Zu Beginn der 2000er-Jahre stellte Virtualisierungssoftware den Alltag vieler Entwickler auf den Kopf: Plötzlich war es möglich, auf einem Rechner Linux *und* Windows auszuführen, unkompliziert Programme in verschiedenen Umgebungen bzw. Web-Apps in alten Versionen von Webbrowsern auszuprobieren, verschiedene Software-Stacks in virtuellen Maschinen parallel zu installieren und zu testen und vieles mehr.

Natürlich spielen virtuelle Maschinen für Entwickler noch immer eine große Rolle; außerdem ist die Cloud in ihrer jetzigen Form ohne Virtualisierung gar nicht denkbar. Dennoch hat vor einigen Jahren ein Umbruch weg von virtuellen Maschinen hin zu Containern begonnen – und dieser Umbruch scheint sich mehr und mehr zu beschleunigen.

Container ermöglichen es, bestimmte Softwarekomponenten (Webserver, Programmiersprachen, Datenbanken) ohne den Overhead einer virtuellen Maschine auszuführen. Warum ein ganzes Betriebssystem (meist Linux) in eine virtuelle Maschine installieren, wenn es doch nur um *eine* ganz spezifische Funktion geht?

Selten trifft das Paradigma »weniger ist mehr« so gut zu wie auf die Container-Technologie. Das »weniger« drückt sich in unzähligen Vorteilen aus: Container sind viel schneller aufgesetzt als virtuelle Maschinen, lassen sich leichter auf verschiedenen Entwicklungssystemen replizieren, beanspruchen weniger Ressourcen und bieten wesentlich bessere Möglichkeiten zur Skalierung und Lastverteilung. Container sind insofern nicht nur ein Segen für Entwicklerteams, sondern bieten auch vollkommen neue Möglichkeiten im Deployment, also im produktiven Betrieb der entwickelten Lösung.

## Docker oder Podman? Beides!

Der Firma Docker ist es mit ihrem gleichnamigen Produkt gelungen, dem Einsatz von Containern zum Durchbruch zu verhelfen. Docker ist aber bei Weitem nicht das einzige Container-System. Die interessanteste Alternative ist das von der Firma Red Hat entwickelte Programm *Podman*. Podman und Docker verfolgen zwar technisch leicht unterschiedliche Ideen, sind aber aus Anwendersicht weitestgehend kompatibel miteinander: Die Optionen zentraler Kommandos sind vollkommen identisch, für wichtige Konfigurationsdateien gilt dieselbe Syntax. Sie können also fast mühelos zwischen Docker und Podman wechseln.

Für Podman sprechen die unter Linux einfachere Installation und das liberalere Lizenzmodell. Während Red Hat das Geld mit seinen Enterprise-Produkten verdient und Podman einfach ein Teil davon ist, gibt es bei Docker diverse kostenpflichtige Zusatzangebote. Insbesondere ist der kostenlose Einsatz der grafischen Benutzeroberfläche *Docker Desktop* nur für Privatanwender und kleine Firmen erlaubt.

Wenn man regelmäßig die Zeitschrift *c't* liest, bekommt man den Eindruck, Docker wäre sowieso schon seit Jahren tot und Podman längst sein Nachfolger. Die alljährlich von Stack Overflow durchgeführte große Umfrage unter allen Entwicklern zeigt das genaue Gegenteil: Mehr als die Hälfte aller professionellen Entwickler setzen Docker ein (57 %), aber nur 4 % Podman (Stand: Juni 2023).

Kleines Detail am Rande: Während Red Hat mit Podman unbestritten eine großartige Alternative zu Docker entwickelt hat, überlässt man das kostenintensive Hosting von Images großzügig der Firma Docker. Oder, anders formuliert: Auch wenn Sie Podman verwenden, werden Sie den Großteil der Images aus dem Docker Hub beziehen. Red Hat betreibt zwar eine eigene Image-Registry, aber dort werden nur Projekte aus dem Red-Hat-Universum gehostet.

Langer Rede kurzer Sinn: Wir behandeln in diesem Buch sowohl Docker als auch Podman. Dennoch gilt für uns Docker als der »Goldstandard«. Wenn Sie mit Podman arbeiten und ein Fehler auftritt, werden Sie sich immer die Frage stellen: Habe ich einen Fehler gemacht? Oder ist Podman schuld? Zumindest bei uns ging die Verteilung in die Richtung 99 % eigene Fehler, 1 % Podman-Inkompatibilität. Aber eine Restunsicherheit bleibt – und die wird mit Internetrecherchen selten besser. In der Vergangenheit gab es unzählige Inkompatibilitäten zwischen Docker und Podman. Auch wenn diese zumeist behoben sind, stoßen Sie natürlich weiterhin auf die diesbezüglichen Fehlerberichte im Internet. Beachten Sie immer das Veröffentlichungsdatum bzw. die angegebenen Versionsnummern!

### Wozu dieses Buch?

In diesem Buch geben wir Ihnen eine Einführung in den Umgang mit Docker und einen Überblick über die wichtigsten Bausteine (*Images*), aus denen Sie eigene Container-Welten zusammensetzen können. Wir zeigen Ihnen anhand mehrerer großer Beispiele, wie Sie Docker in der Praxis einsetzen, und gehen ausführlich auf das Deployment in der Cloud ein.

Wir haben das Buch in drei Teile gegliedert:

- **Teil I** stellt Docker vor. Sie lernen anhand vieler Beispiele, wie Sie die Kommandos `docker` bzw. `podman` sinnvoll einsetzen und wie die Syntax der Dateien `Dockerfile` und `compose.yaml` aussieht.

- ▶ **Teil II** präsentiert wichtige Images, die als Basis für eigene Projekte dienen können. Dazu zählen unter anderem:
  - Alpine Linux
  - die Webserver Apache, Nginx und Caddy (inklusive Proxy-Setup mit Traefik sowie Let's-Encrypt-Konfiguration)
  - die Datenbankserver MySQL/MariaDB, MongoDB, PostgreSQL und Redis
  - die Programmiersprachen Go, JavaScript (Node.js), PHP, Python und Ruby
  - die Webapplikationen WordPress, Joomla und Nextcloud
- ▶ **Teil III** zeigt den Einsatz von Docker in der Praxis. Wir zeigen Ihnen sowohl, wie Sie moderne Webapplikationen mit Docker besonders effizient entwickeln, als auch, wie Sie vorhandene Projekte mit all ihren Altlasten in besser wartbare Docker-Projekte umwandeln.

Zwei Kapitel zur Nutzung von GitLab mit Docker und zu *Continuous Integration* (CI) und *Continuous Delivery* (CD) demonstrieren Ihnen neue Paradigmen und Hilfsmittel für das Entwickeln von Software im Team.

Auch das Deployment werden wir nicht vernachlässigen: Mit Docker *Swarm* und *Kubernetes* bringen Sie Ihre Docker-Projekte in die Cloud und profitieren von den dort gegebenen Möglichkeiten zur Skalierung. Eine Sammlung von Tipps stellt sicher, dass dabei die Sicherheit nicht zu kurz kommt.

### Neu in der vierten Auflage

Für diese Auflage haben wir das Buch vollständig aktualisiert und das Installationskapitel mit seinen unzähligen Varianten übersichtlicher strukturiert. Wichtige inhaltliche Neuerungen sind:

- ▶ **mehr Podman-Beispiele und -Details**
- ▶ **Docker und Podman Desktop:** neue GUIs zur Container-Administration
- ▶ **Multi-Architecture-Builds:** Images für Intel- und ARM-CPU's
- ▶ **Caddy:** Nutzung des HTTPS-only-Webservers als Container
- ▶ **Go:** kleines REST-Beispiel

### Schöne neue Container-Welt

Wer Docker oder Podman einmal ausprobiert und kennengelernt hat, will nie wieder auf seine Funktionen verzichten. Lassen Sie sich von uns in eine neue Welt führen!

Bernd Öggl (<https://webman.at>)  
Michael Kofler (<https://kofler.info>)

## Materialien zum Buch

Auf der Webseite zu diesem Buch stehen folgende Materialien für Sie zum Download bereit:

- ▶ alle Projektdateien
- ▶ alle Codebeispiele

Gehen Sie auf <https://www.rheinwerk-verlag.de/5742>. Klicken Sie auf den Reiter MATERIALIEN ZUM BUCH. Dort sehen Sie eine Liste der herunterladbaren Dateien samt einer Kurzbeschreibung. Klicken Sie auf den Button HERUNTERLADEN, um den Download zu starten.

## GitHub

Wir verwenden Git zur Verwaltung unserer Beispieldateien. Deshalb finden Sie (eventuell bereits aktualisierte oder korrigierte) Beispieldateien auch auf GitHub:

<https://github.com/docker-compedium/docker4-samples>

Das Repository enthält eine Übersicht der einzelnen Kapitel mit den Links zu den entsprechenden Repositories mit Git-Sub-Modulen. Das bedeutet, dass Sie mit den folgenden Befehlen alle Beispiele von GitHub auf Ihren Computer laden können:

```
git clone \  
  https://github.com/docker-compedium/docker4-samples.git
```

Sollten Sie Verbesserungsvorschläge haben, freuen wir uns natürlich über Pull-Requests von Ihnen.

## Testplattformen und Linux-Distributionen

Unsere Haupttestplattform für dieses Buch war Ubuntu Linux. Parallel dazu haben wir viele Beispiele auch unter Windows und macOS sowie unter Debian, Fedora und AlmaLinux ausprobiert.

Noch eine Anmerkung zu Red Hat: Wenn in diesem Buch von RHEL die Rede ist, dann ist damit nicht nur Red Hat Enterprise Linux in der Version 9 gemeint; vielmehr gelten die Informationen auch für alle damit kompatiblen Distributionen. Das gilt insbesondere für AlmaLinux, CentOS Stream, Rocky Linux und Oracle Linux.