

# Cloud Service Provider Evaluierung auf Basis von Infrastructure as Code Unterstützung

Julian Schallmueller

Kai Warendorf

Fakultät Informatik & Informationstechnik Hochschule Esslingen – University of Applied Sciences

Arbeit durchgeführt bei der Firma Novatec Consulting GmbH, Leinfelden-Echterdingen

## Einleitung

Cloud Computing ist heute in vielen Bereichen ein fester Bestandteil des Alltags. Neben bekannten Cloud Storage Angeboten wie Dropbox und OneDrive findet auch eine wachsende Anzahl an cloudbasierten Services in Form von Microsoft Office 365, Netflix und weiteren gleichermaßen im Business wie auch im privaten Bereich eine immer breitere Verwendung. Das Service Modell Software as a Service (SaaS) stellt sowohl den am stärksten sichtbaren als auch größten Bereich dar, gefolgt von Infrastructure as a Service (IaaS) an zweiter Stelle [2]. Befeuert durch die Auswirkungen der COVID-19-Pandemie erfährt besonders das IaaS Segment ein enormes Wachstum, [4] [2] der Aufbau von entsprechenden Kompetenzen und der Einsatz effizienter Arbeitsprozesse, Methoden und Tools wird daher immer wichtiger.

## Motivation und Zielsetzung

Wenn eine Organisation sich dazu entscheidet bestehende oder zukünftige Infrastruktur in die Cloud zu migrieren ist es entscheidend auch passende Methoden anzuwenden, Arbeitsprozesse die sich bisher bewährt haben sind oftmals ineffektiv und verhindern das Ausnutzen des vollen Potentials der Cloud [3]. Wird dementsprechend die Entscheidung getroffen das Infrastruktur Provisioning und Management nach Infrastructure as Code (IaC) Prinzipien vorzunehmen ist Terraform wahrscheinlich für die meisten Organisationen das Tool der Wahl, es ist in diesem Bereich aktuell das am weitesten verbreitete Tool. Zu dem Zeitpunkt zu dem entschieden wird Terraform einzusetzen hat eine Organisation in der Regel bereits existierende Systeme und Services die auf einer Cloud Plattform gehostet werden. Die drei größten Plattformen für Cloud Computing sind Amazon AWS, Microsoft Azure und Google Cloud Platform, daher ist das Ziel dieser Arbeit die Unterstützung von Terraform dieser drei Anbieter zu betrachten und zu vergleichen. Die Entscheidung ob die aktuell verwendete Plattform

für die Implementierung von IaC gut geeignet ist oder ob ein Wechsel sinnvoll ist soll durch die Ergebnisse der Arbeit unterstützt werden.

## Infrastructure as Code mit Terraform

Terraform verwendet eine deklarative Domänenspezifische Sprache die sowohl Maschinenfreundlichkeit als auch Menschenlesbarkeit vereint. Der grundlegende Terraform Workflow besteht aus vier Befehlen: *init*, *plan*, *apply* und *destroy*. Der *terraform init*-Befehl installiert die notwendigen Plugins, die Terraform Provider, die zur Interaktion mit der Zielplattform wie zum Beispiel AWS, Azure oder GCP benötigt werden und konfiguriert das Terraform Commandline-Interface. Mit *terraform plan* wird ein Execution Plan erzeugt der ermittelt welche Abhängigkeiten zwischen den Elementen des Infrastruktur-Systems bestehen und unter Berücksichtigung bereits existierender Teile festlegt welche Schritte vorgenommen werden müssen um den im Code definierten Soll-Zustand zu erreichen. Die Ausführung eines Execution Plans wird dann durch *terraform apply* angestoßen. Ein nicht mehr benötigtes System kann durch *terraform destroy* wieder vollständig abgebaut werden. Ein Vorteil den Terraform bietet ist das Strukturieren des Infrastruktur Codes in Module die in der Regel eine zusammengehörige Gruppe von Ressourcen umfassen, zum Beispiel alle Ressourcen die für ein Netzwerk in Microsoft Azure benötigt werden. Solche Module werden von der Community in einer öffentlichen Public Module Registry zur freien Verwendung bereitgestellt, grob vergleichbar mit der npm Registry für Nodejs.

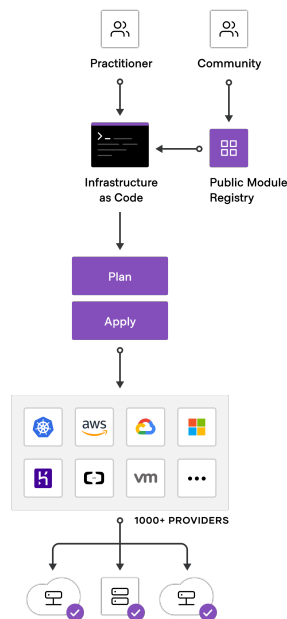


Abb. 1: Terraform Funktionsprinzip [5]

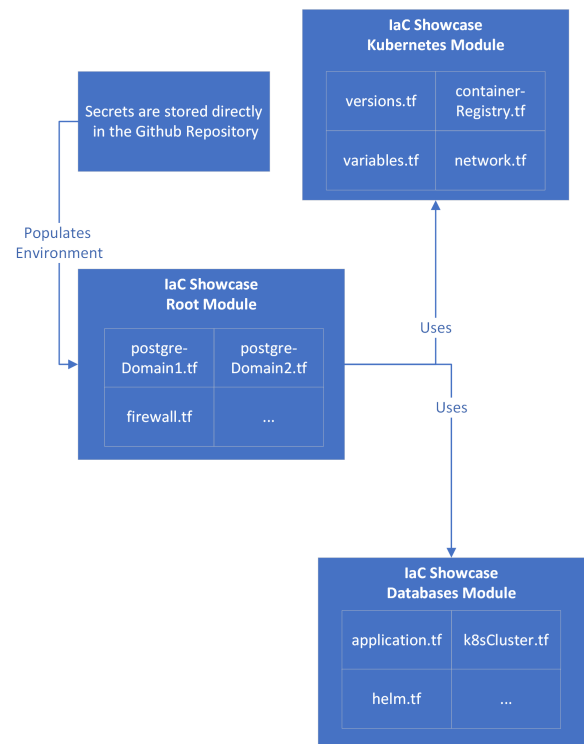


Abb. 2: High-Level Struktur des zu untersuchenden Systems [1]

Die konkrete Implementierung unterscheidet sich im Detail von der Darstellung, diese soll primär dazu dienen eine Orientierung zu bieten wie die Implementierung ausgeführt werden könnte. Was jedoch beibehalten wird ist die Strukturierung der Module.

## Auswahl der Vorgehensweise und Vergleichskriterien

Um einen möglichst gut strukturierten und objektiven Vergleich erreichen zu können soll nach dem in der ISO/IEC 25040 beschriebenen Prozess werden. Dieser besteht aus fünf Schritten: Im ersten Schritt werden die Anforderungen an den Vergleich festgestellt. Im zweiten Schritt werden die Kriterien spezifiziert. Der dritte Schritt befasst sich mit der Planung der Evaluierung die dann im vierten Schritt durchgeführt wird. Die letzte Phase ist die Auswertung und Bewertung der Ergebnisse. Als Grundlage der Vergleichskriterien soll das Software Qualitätsmodell der ISO/IEC 25010 dienen.

## Ausblick

Im Vorfeld ist zu erwarten dass die drei untersuchten Cloud Service Provider AWS, Azure und Google Cloud Plattform sich in etwa gleichermaßen für den Einsatz in einem gewöhnlichen Szenario eignen. Es

## Aufbau des zu vergleichenden Systems

Das System das untersucht werden soll besteht aus typischen Computer Ressourcen die in vielen System zum Einsatz kommen. Dazu gehören virtuelle Maschinen, Netzwerk, Datenbanken, Storage und ein Kubernetes-Cluster.

wäre daher interessant den Vergleich in Zukunft auf weitere Anbieter zu erweitern, Alibaba, Oracle, IBM und Tencent bieten die nächst größten Plattformen für Cloud Computing an. Ein ebenfalls relevantes Thema ist der Vergleich von Terraform mit anderen IaC Tools.

Besonders von Interesse wäre ein Vergleich zwischen Terraform und Pulumi, einem Tool das in etwa die selbe Funktion wie Terraform erfüllt, jedoch deutlich jünger und daher noch weniger stark verbreitet ist.

## Literatur und Abbildungen

- [1] Eigene Darstellung.
- [2] Anders Lisdorf. *Cloud Computing Basics: A Non-Technical Introduction*. Apress Media LLC, 2021.
- [3] Kief Morris. *Infrastructure as Code*. O'Reilly Media, 2 edition, 2020.
- [4] ohne Verfasser. Gartner says four trends are shaping the future of public cloud. <https://www.gartner.com/en/newsroom/press-releases/2021-08-02-gartner-says-four-trends-are-shaping-the-future-of-public-cloud>, 08 2021.
- [5] ohne Verfasser. How terraform works. <https://www.terraform.io/>, 2021.