

Cisco HyperFlex im Einsatz



Hyperkonvergente Infrastruktur

Markus Stubbig

Inhaltsverzeichnis

Vorwort

1. Einleitung

- Wie war es vorher?
- Konvergent
- Hyperkonvergent
- Pro und Contra
- Cisco HyperFlex
- Zusammenfassung

2. Architektur

- Wie ist HyperFlex aufgebaut?
- Komponenten
- Netzwerk
- Lizenzierung
- Adressen
- Planung
- Labor
- Zusammenfassung

3. Installation

- Fabric Interconnect
- UCS-Manager
- UCS-Server

- vCenter
- Versionen
- HyperFlex Installer
- Kontrolle
- Zusammenfassung

4. **Einrichtung**

- HyperFlex Connect
- Die erste VM entsteht
- Updates
- Zusammenfassung

5. **Ausfallschutz**

- Datenträger
- Server
- Fabric Interconnect
- Zusammenfassung

6. **Migration**

- VMware Converter
- xvMotion
- Zusammenfassung

7. **Sicherheit**

- Managementzugang
- VLANs und vSwitches
- Logging
- Zertifikate
- Zusammenfassung

8. **Einblick**

HyperFlex Installer
Storage Controller
Zusammenfassung

9. **Überwachung**

Monitoring
Backup und Restore
Zusammenfassung

10. **Best Practice**

Cluster erweitern
Zugriff auf den Datastore
Leistungsmessung
Sizing
Neues VLAN
Lokations-LED

11. **Ausblick**

Stretched Cluster
Replizierung
Edge Cluster
Verschlüsselte Datenträger

12. **Literaturverzeichnis**

13. **Stichwortverzeichnis**

A. **Zusatzmaterial**

B. **Programmierschnittstelle**

Vorwort

Konvergent, hyperkonvergent, wer bietet mehr? Was hier nach tiefer, breiter, schneller klingt, ist keine neue Wunder-Appliance, sondern eine Mischung aus Effizienz und Rückschritt. Das Konzept dahinter ist einfach: weg von komplexen Storage-Arrays, verschachtelten Netzstrukturen, wilden Verkabelungen und Hersteller-Mix. Zurück zum lokalen Storage und zwei Netzanschlüssen pro Server.

Die Anbieter von hyperkonvergenten Produkten versprechen den potenziellen Kunden, dass alles einfacher wird. Ob das stimmt oder nicht verrät ein Blick in ein hyperkonvergentes Mini-Rechenzentrum: zwei Switches und drei Server. Insgesamt fünf Höheneinheiten. Und es ist vollständig! Nach ein paar Stunden der Planung, Verkabelung und Einrichtung ist das Ding einsatzbereit.

Das Konzept kommt am Markt gut an: Die Kunden fühlen sich mit simpler IT wohl und bescheren den Herstellern neue Umsätze. Und Gartner spendiert dem Geschehen sogar einen eigenen Quadranten für *Hyperconverged Infrastructure*. Cisco platziert sich mit HyperFlex im rechten, oberen Block bei den „Gewinnern“. Zugegeben: Es gibt noch Abstand zum Marktführer und zu den anderen Storage-Profis.

Aber wie werden zwei Switches und drei Server hyperkonvergent? Im Hintergrund wandert mehr Intelligenz in die Software. Die Hardware degradiert sich zur Standardhardware. Im Rack wohnen handelsübliche x86-Server und darin stecken normale Festplatten oder Solid-State-Laufwerke. Die Software kümmert sich um die Details

und komponiert ein Dateisystem über alle Disks. Flott und robust.

Die Zeit wird zeigen, ob HyperFlex die Rechenzentren überschwemmen wird oder nur ein modischer Trend ist. Aber wenn das erste hyperkonvergente System an die eigene Tür klopft, wird es Zeit für fundiertes Grundwissen. Falls die offizielle Dokumentation von Cisco zu langatmig ist, liegt die Kurzform gebunden oder als E-Book vor Ihnen: Denn dieses Buch will einen schnellen Einstieg in HyperFlex liefern und die Besonderheiten vorführen, die mit der Lösung möglich sind.

Viel Spaß beim Ausprobieren, Staunen und Fluchen.

Übersicht

[Kapitel 1](#) beschreibt den Wandel im Rechenzentrum von gewachsenen Strukturen bis zur hyperkonvergenten Infrastruktur. Als anschauliches Produkt erklärt [Kapitel 2](#) die Architektur von HyperFlex und die enthaltenen Komponenten. Noch konkreter wird das Beispielnetz, welches eine reale HyperFlex-Umgebung plant. Aus der Planung wird in [Kapitel 3](#) Wirklichkeit durch Vorbereitung der Server und Switches. Danach darf der Installer loslegen und daraus ein HX-Cluster formen.

Die neue Umgebung nimmt [Kapitel 4](#) unter die Lupe und spielt mit Datastores, virtuellen Maschinen und Updates. Bevor der erste Workload startet, fallen in [Kapitel 5](#) Festplatten, Server, Netzteile und Switches aus, ohne dass die Umgebung ernsthaft Schaden nimmt.

Mit dem frisch gewonnenen Vertrauen beginnt in [Kapitel 6](#) die Migration einer bestehenden Umgebung auf das HyperFlex-Cluster. Parallel dazu klopft [Kapitel 7](#) die Sicherheit ab und liefert hier und da einen Security-Tipp.

Damit schnurrt die neue Hardware, aber [Kapitel 8](#) bohrt etwas tiefer und erklärt, wie das software-definierte Speichernetz von HyperFlex unter der Oberfläche arbeitet. Für den regulären Betrieb sichert [Kapitel 9](#) die Konfiguration und überwacht die Server per Monitoringsystem auf Ausfälle.

Danach stellt [Kapitel 10](#) die HyperFlex-Umgebung auf den Prüfstand und verteilt Tipps für die Dimensionierung einer neuen Umgebung. Zuletzt gibt [Kapitel 11](#) einen Ausblick auf größere und kleinere Cluster.

Ressourcen

Der beste Einstiegspunkt für Dokumentation rund um HyperFlex und UCS-Server ist die *Cisco HyperFlex Systems Documentation Roadmap* [1]. Hier tummeln sich Anleitungen, Checklisten, Admin-Guides und Release Notes zu allen Hardwarekomponenten und Softwareständen.

Für die visuellen Lerner gibt es bereits eine beachtliche Auswahl an Videos auf YouTube. Das Spektrum reicht von kurzen Werbeclips, Diskussionen zwischen Experten bis zum vollständigen Installationsvideo.

Für den Praktiker stellt Cisco kostenlos eine simulierte HyperFlex-Demo-Umgebung in der *dCloud* bereit. Dort können Interessierte in der vorgegebenen Topologie mit den Komponenten experimentieren. Für den Einstieg eignet sich das Szenario „Installation and Management Lab“.

Schriftkonventionen

Nichtproportionalschrift zeigt die erzeugte Ausgabe eines Kommandos.

Schreibmaschinenschrift wird für Konfigurationen und Schlüsselwörter benutzt, die buchstabengetreu eingetippt werden müssen.

Nichtproportionalschrift Fett zeigt Befehle, die eine Ausgabe erwarten.

Hervorhebungen weisen auf besondere Wörter oder Zeilen innerhalb von Kommandos oder Bildschirmausgaben hin.

```
ein-sehr-langer-kommando-aufruf --mit --sehr \  
--vielen "Optionen"
```

Kommandos mit vielen Argumenten können länger als eine Zeile sein. Für die bessere Übersicht werden diese Kommandos mehrzeilig abgedruckt und um zwei Zeichen eingerückt. Am Ende jeder Zeile steht der Backslash als Hinweis darauf, dass es in der nächsten Zeile weitergeht.

Rechtliches

Warennamen und Bezeichnungen werden ohne Gewährleistung der freien Verwendbarkeit benutzt. Es ist davon auszugehen, dass viele der Warennamen gleichzeitig eingetragene Warenzeichen oder als solche zu betrachten sind.

Bei der Zusammenstellung von Texten, Bildern und Daten wurde mit größter Sorgfalt vorgegangen. Trotzdem können Fehler nicht vollständig ausgeschlossen werden. Der Autor lehnt daher jede juristische Verantwortung oder Haftung ab. Für Verbesserungsvorschläge und Hinweise auf Fehler ist der Verfasser dankbar.

Icons

Die Diagramme enthalten Symbole vom *Icons8* (<https://icons8.com>) und *VMware*. VMware erlaubt die Benutzung ihrer Icons in Diagrammen, wenn die Nutzungsbedingungen abgedruckt sind:

This document was created using the official VMware icon and diagram library. Copyright © 2012 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at <http://www.vmware.com/go/patents>.

VMware does not endorse or make any representations about third party information included in this document, nor does the inclusion of any VMware icon or diagram in this document imply such an endorsement.

Kapitel 1

Einleitung

Eine hyperkonvergente Infrastruktur (HCI) bringt Rechenleistung (*compute*), Speicherplatz (*storage*) und Netzwerk (*networking*) zusammen und packt sie in ein einzelnes IT-System. HCI implementiert die drei Säulen in Software und ersetzt damit die bisherigen hardware-basierten Systeme. Das Ergebnis ist eine Plattform mit Hypervisor, software-definiertem Storage und virtualisiertem Netzwerk. Die Software läuft auf Standard-Servern und nutzt reguläre Speichermedien.

Wie war es vorher?

HCI ist ein recht neues Konzept und die Vorteile werden deutlicher, wenn man den Wandel im Rechenzentrum betrachtet. Zwei Evolutionsschritte vorher, im traditionellen Rechenzentrum, liefen die Applikationen auf dedizierten Servern. Jeder Server hat mehrere Verbindungen ins lokale Netzwerk und mehrere Zugänge ins Speicher-Netz. Die Anwendungen legen ihre Daten nicht lokal ab, sondern auf Festplatten im Storage-Netzwerk. Die drei Bereiche *compute*, *storage* und *networking* sind separate Einheiten, physikalisch getrennt und werden meist von unterschiedlichen Teams betreut. [Abbildung 1.1](#) auf der nächsten Seite zeigt den traditionellen Ansatz.

Je nach Anbieter, Modell und Konfiguration zeigt sich ein operatives Problem, denn das Zusammenspiel der Einheiten ist nicht garantiert. Im Fehlerfall sind mehrere Hersteller involviert und die potenzielle Ursache wird gerne hin-und-her geschoben.

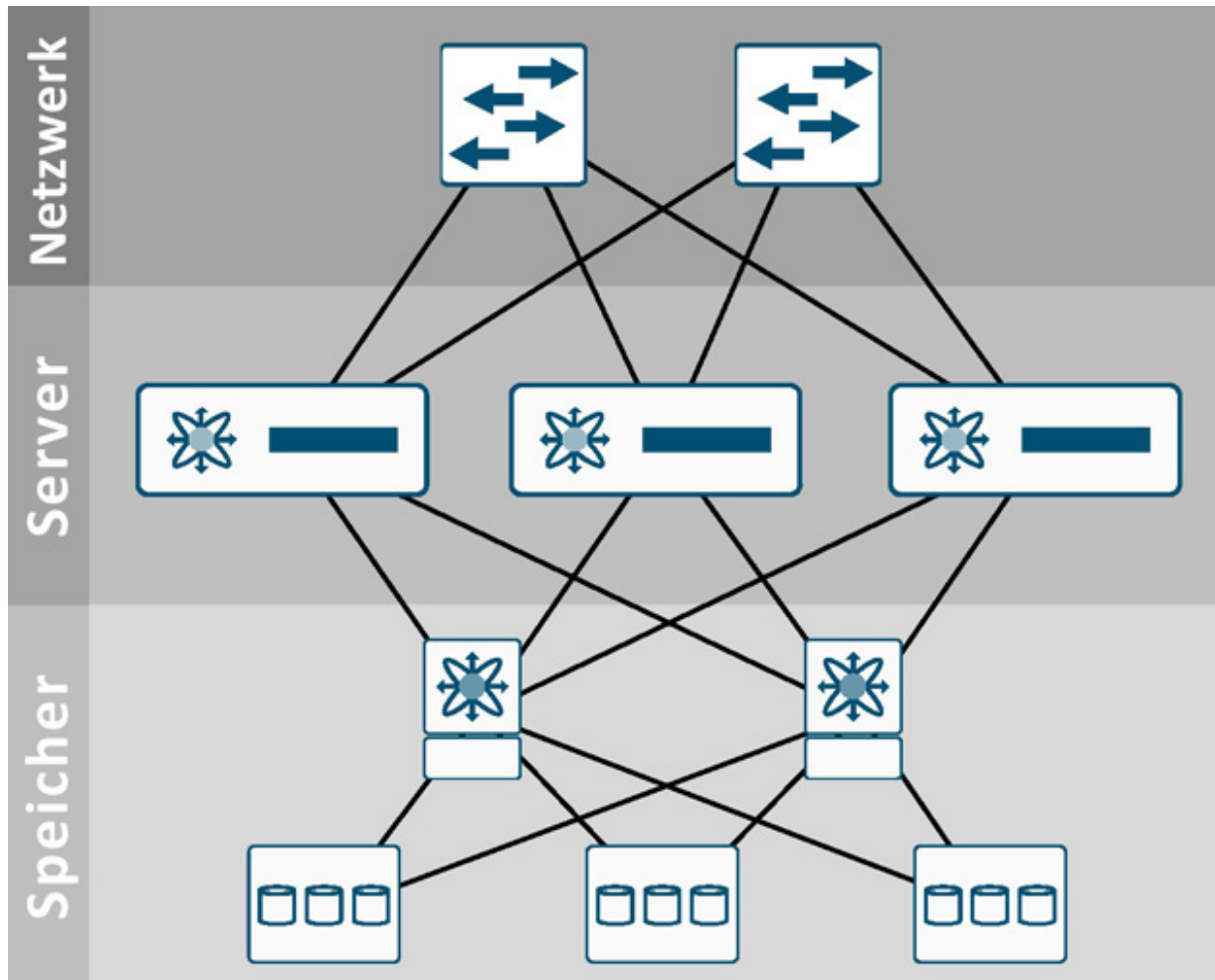


Abbildung 1.1: Der traditionelle Aufbau unterteilt in Speicher, Server und Netzwerk

Und selbst ein funktionierendes System kann nach einem Update wackeln, denn Softwareupdates sind komplex, da es viele Abhängigkeiten gibt. Die Zeitspanne von der Planung bis zur Inbetriebnahme liegt bei Monaten.

Vor dem Hintergrund der traditionellen Architektur sind erste Trends zum Zusammenspiel erkennbar. Hersteller testen Interoperabilität und veröffentlichen ihre Ergebnisse als Referenzdesign oder Aufbauanleitung (Beispiel: Cisco FlexPod).

Konvergent

In der konvergenten Infrastruktur sind Compute, Storage und Netzwerk immer noch separate Einheiten, aber von *einem* Partner zusammengestellt. Der Anbieter verkauft sein Produkt als Gesamtpaket. Der Vorteil: Alle Bauteile sind gegeneinander getestet und validiert.

Um ein einheitliches Gesamtpaket zu verdeutlichen, legt der Anbieter eventuell eine Managementsoftware bei, die alle Komponenten zusammenfasst und dem Anwender präsentiert.

Beim Softwareupdate schreibt der Anbieter die Versionen aller Bauteile vor, um die Stabilität weiterhin zu gewährleisten. Natürlich bleiben Fehler nicht aus, aber im Supportfall gibt es nur noch *eine* Nummer zum Anrufen.

Die Zeitspanne von der Planung zur Inbetriebnahme liegt bei Wochen (Beispiel: VCE VBlock).

Hyperkonvergent

Der hyperkonvergente Ansatz fasst Compute und Storage zusammen und baut sie in die Server ein. Eine zusätzliche Software verwaltet den Storage in den Servern. Diese Software weiß genau, welche Festplatte in welchem Server den angefragten Datenblock gespeichert hat. Ein separates SAN gibt es nicht. Das Ergebnis ist ein *Software-defined Storage*.

Die *Hyperkonvergente Infrastruktur* (HCI) ist das harmonische Zusammenspiel von Serverhardware mit der HCI-Software. Grundsätzlich können Hardware und Software von unterschiedlichen Herstellern kommen. Bei *HyperFlex* kommt beides von Cisco. Das Ergebnis ist eine perfekte Optimierung der Software an die verwendete Hardware. Der Support aller Komponenten kommt aus einer Hand. Die Zeitspanne von der Planung bis zur Inbetriebnahme liegt bei Tagen.

Pro und Contra

Weder HCI noch HyperFlex ist die ultimative Allzweckwaffe gegen jede Art von Problemen im Rechenzentrum. Allerdings deckt es viele Workloads ab und vereinfacht die Administration.

Vorteile

Die hyperkonvergente Infrastruktur benötigt weder spezielle Server, komplizierte Storage-Systeme oder Spezialwissen. Die verwendeten Server teilen die Arbeit unter sich auf. Der Storagecontroller verteilt die Datenblöcke auf alle Festplatten und hat den Füllstand stets im Blick. Das Ergebnis ist eine bessere Auslastung aller Ressourcen.

Wenn die verfügbare Rechenleistung oder Speichermenge schwindet, lässt sich der Engpass durch zusätzliche Server beheben. Jede weitere Maschine beschert dem HCI-Cluster ein bisschen Compute und ein bisschen Storage. Das Konzept skaliert, bis es an die festgelegte Grenze von 64 Servern stößt. Weiterhin versprechen die Marketingbroschüren eine Reduktion der Datenmenge durch Deduplizierung (20-30%) und Komprimierung (30-50%). Die Weboberfläche von HyperFlex (vgl. [Kap. 4](#)) wird zeigen, wie hoch die Einsparungen tatsächlich sind.