

Advanced Developers Conference

Development for Professionals!

21

Monolith als Design Pattern

Eine Reise durch das Tal der Monolithen

Stephan Hüwe



Über mich

- Diplom-Informatiker (FH)
- Softwareentwickler (.NET 1.1 / 2003), Trainer, Autor
- Lehrbeauftragter an der Hochschule Augsburg
- C#, IoT, Blockchain



Ihr Reiseplan

Inhalt

Einführung
Monolith
Microservice als Gegensatz
Lösungswege
Diskussion und Fazit



Motivation

- Balance schaffen
- Gegensätzliche Standpunkte
- Softwareentwicklung als Selbstzweck
- Hinterherlaufen
- Weg von Begrifflichkeiten
- Die ersten kommen enttäuscht zurück



Ohoh - Er hat Monolith gesagt

- **Monolith** (altgriechisch μονόλιθος monólithos, deutsch ‚einheitlicher Stein‘ oder wörtlich ‚Ein-Stein‘, aus μονο- mono- ‚einzel-‘ und λίθος líthos ‚Stein‘) ist eine eher allgemeinsprachliche Bezeichnung für einen natürlich entstandenen oder bearbeiteten, aus nur einer Gesteinsart bestehenden Gesteinsblock. Aus Wikipedia



Live Demo



Wer entwickelt (immer) noch so?



Wer entwickelt (immer) noch so?

- Schnell
- Effizient
- Perfekte Maßanzugs-Lösung
- Viele Komfortfunktionen

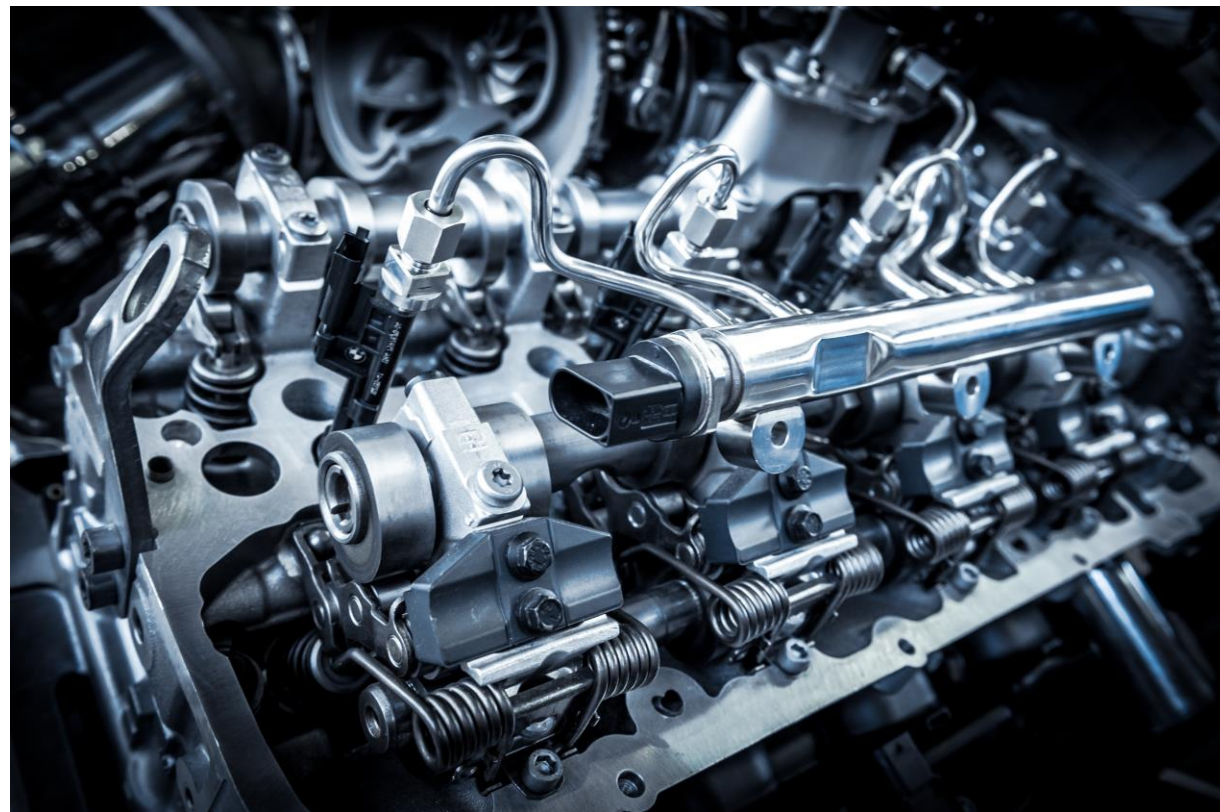


<https://github.com/jilleJr/NackademinUppgift04>



Alles greift perfekt ineinander

- Spezialisten erforderlich
- „Tief unter der Motorhaube“
- Wiederverwendung bedingt
- Wartung schwierig
- Lange Kompilierzeiten
- Uncool

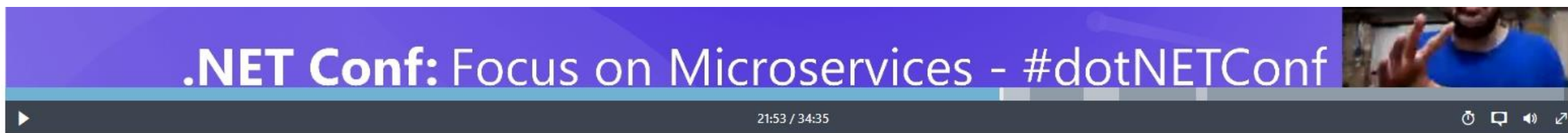


Wir machen jetzt Microservices (2016?) !

- Vorteil?
 - Aber das ist cool !
 - "Do One Thing and Do It Well"
- Großer Baukasten
- Kleine wartbare Elemente
- Alle Sprachen sind erlaubt
- Wiederverwendbarkeit
- Modular
- Damit wird alles besser – versprochen !



Eine Erklärung auf der .NET Conf 2020



SESSION

Evolving .NET Framework Monoliths with .NET 5 and Kubernetes

with Elton Stoneman

Focus on Microservices

Jul 23, 2020

You want to migrate your .NET monolith to microservices so you can run it in Kubernetes? That could be a year-long project, and you really don't need to do it.

This session gives you a pragmatic approach to evolving your monolith, breaking features out into new microservices to get all the benefits of Kubernetes without a full rewrite. We'll start by running a legacy app in Windows containers and end with a distributed app running across Linux and Windows containers in the same Kubernetes cluster.

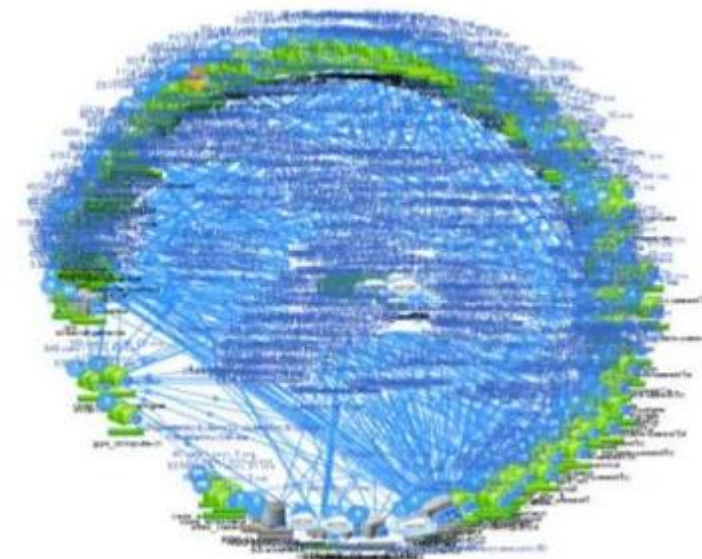
- <https://docs.microsoft.com/en-us/events/dotnetconf-focus-on-microservices/evolving-net-framework-monoliths-with-net-5-and-kubernetes>



„Die Großen“ machen es uns doch vor

- Netflix
- <https://www.slideshare.net/BruceWong3/the-case-for-chaos>

Netflix & Micro-Services



@bruce_m_wong



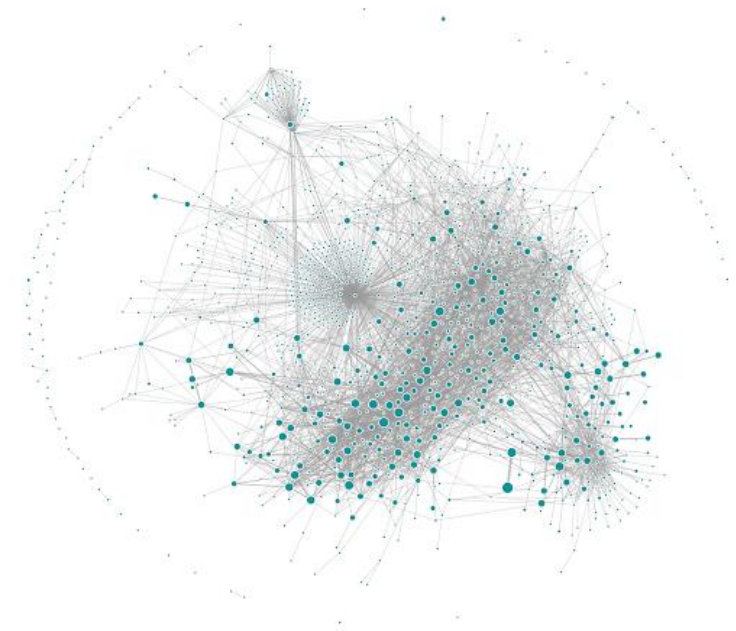
„Die Großen“ machen es uns doch vor

- Twitter
- <https://twitter.com/adrianco/status/441883572618948608>



„Die Großen“ machen es uns doch vor

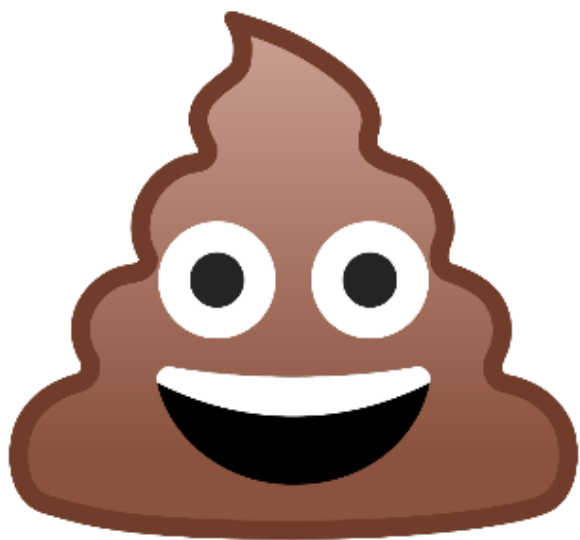
- Uber
- <https://eng.uber.com/microservice-architecture/>



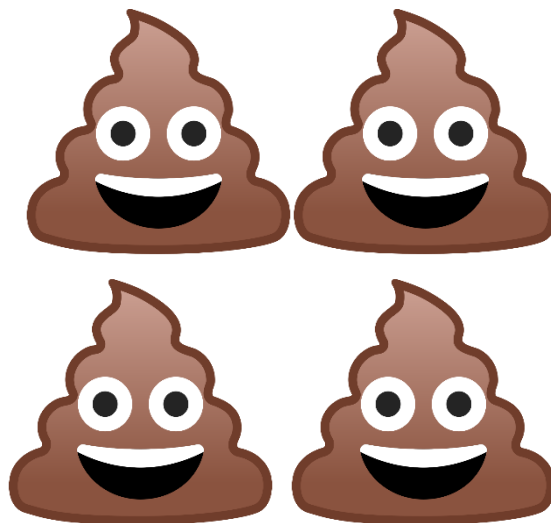
ABER ...



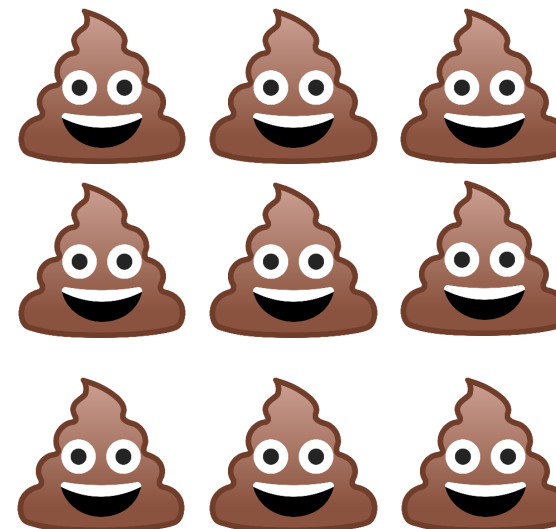
Warum sollte es dieses Mal besser laufen?



Monolith



SOA



Microservices



Live Demo



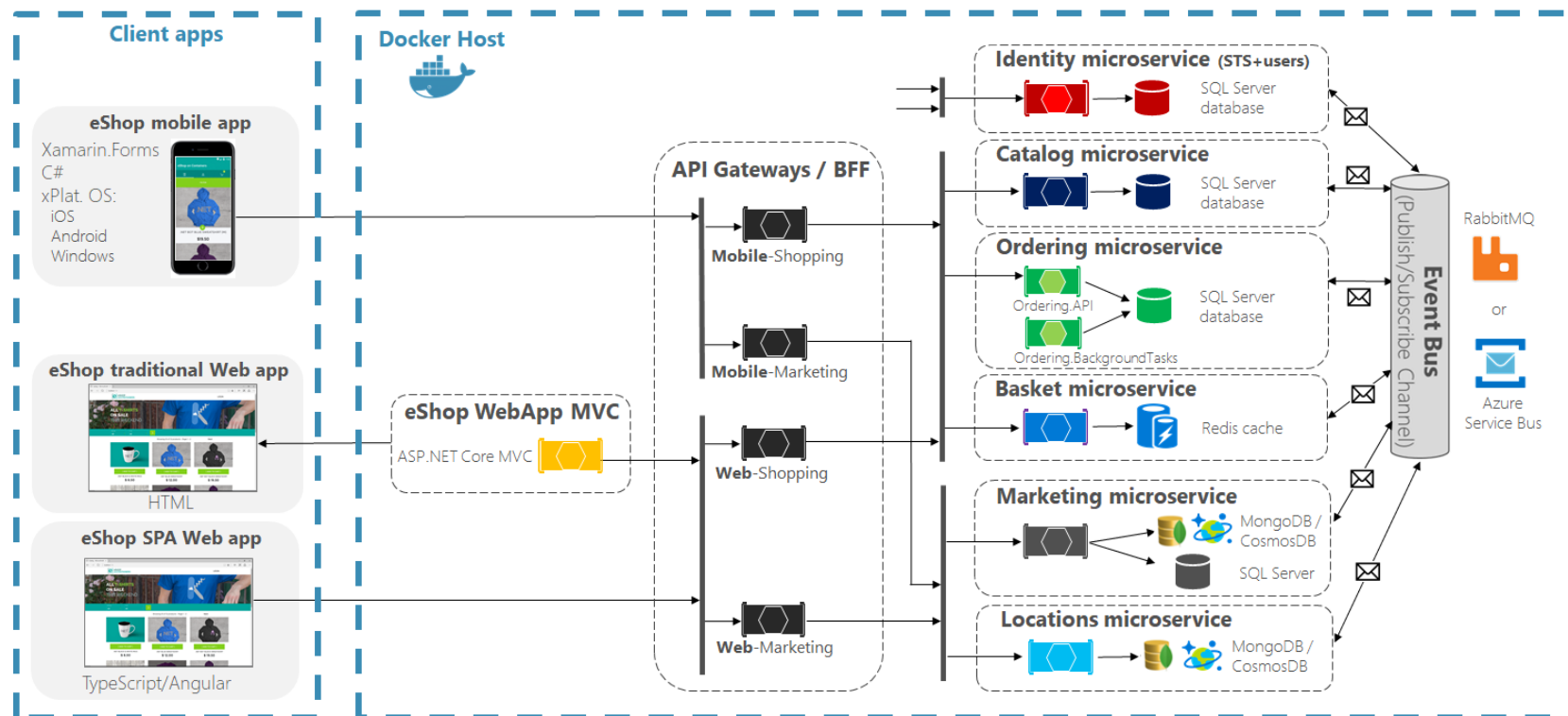
E-Shop on Containers



eShopOnContainers

eShopOnContainers reference application
(Development environment architecture)

<https://docs.microsoft.com/de-de/dotnet/architecture/cloud-native/introduce-eshoponcontainers-reference-app>

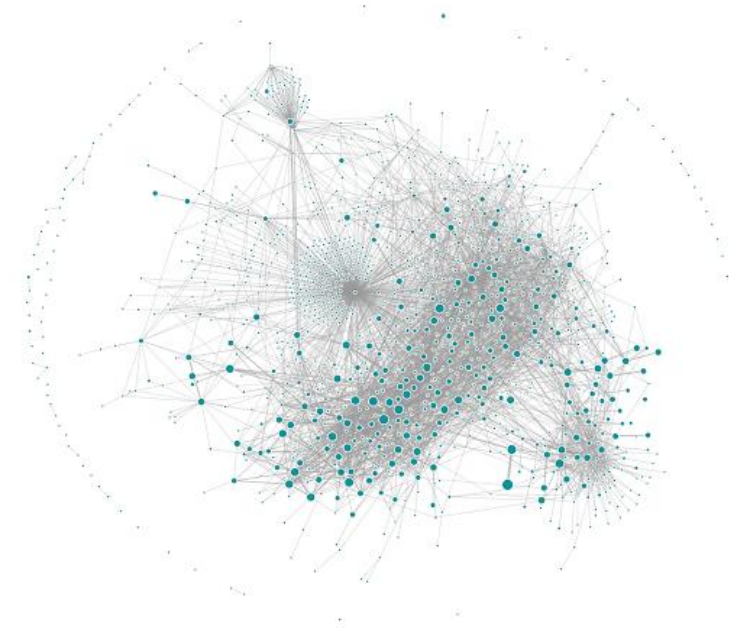


<https://github.com/dotnet-architecture/eShopOnContainers>



Rechenbeispiel

- Wie schnell kann es sein?
 - 100 Aufrufe
 - 100 Millisekunden
 - ➔ 10 Sekunden für eine Funktion?
- Verfügbarkeit?
 - Wenn nur ein Dienst nicht verfügbar ist
 - ➔ 0% Verfügbarkeit
- Works on my machine !



You are not Netflix

- Tausche Maßanzug gegen Klamottenkiste
 - Kubernetes, Docker
 - Event Bus (RabbitMQ / Azure Service Bus)
 - MongoDB, Redis, SQL, NoSQL
 - Cloud
 - DDD
 - Alles rein, was wir kennen?
- Aber was war eigentlich wirklich schlecht?
 - Kleiner wird es nicht
 - Schneller auch nicht
 - Weniger komplex auch nicht



You are not Netflix

- Wer betreut die Architektur?
 - Und wie lange?
- Wie viele Entwickler haben wir?
 - 4 Entwickler, 100 End-User
- Welche Expertise haben wir?
 - ADC
- Sicherheit?
 - Ein Projekt absichern
- Maintenance?
 - DevOps?
- Testen?



You are not Netflix

- CI/CD?
- Bereitstellungskosten?
- DSGVO?
- Vendor Lock?
- Exit Strategie?
- Logging über Services hinweg
 - ELK Stack



Klimawandel voraus

- 1,2,3 Server vs. ?
- Stromverbrauch



Lösungswege ...



Microservices

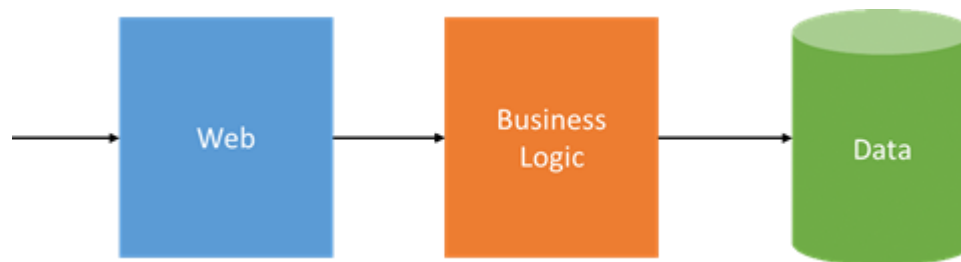


Monolith First ist Pflicht

- Analyse
 - Wo kann es uns einen richtigen Nutzen bringen?
- Master of „eShopOnContainers“
- Ein Green-Field Monolith
 - Will das Produkt wirklich jemand haben?
 - Multiple Lernkurven
- Nur wenn es wirklich auch einen Nutzen bringt
- Kein Selbstzweck



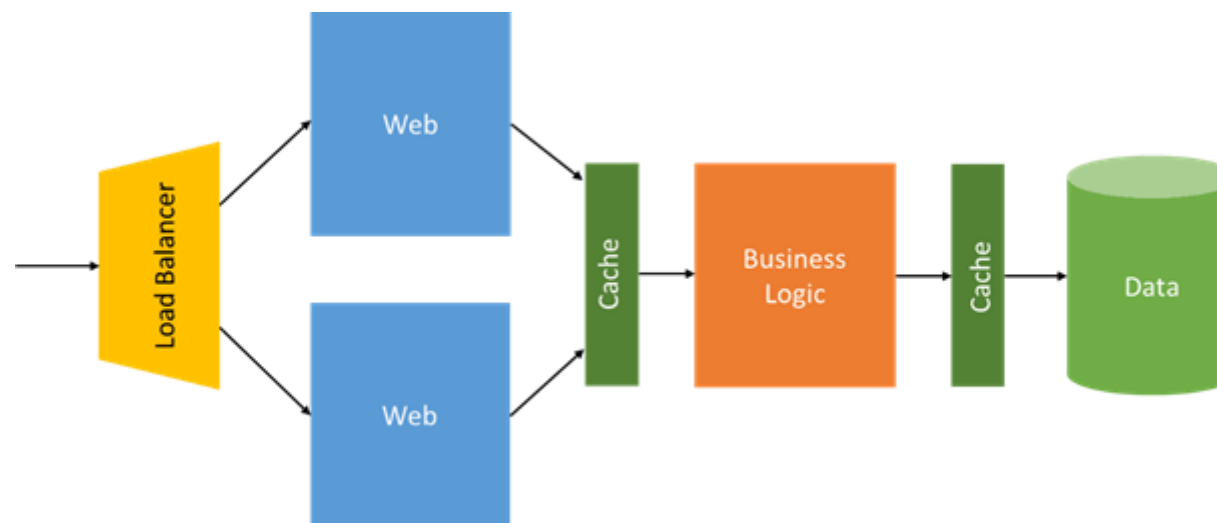
Der Microsoft - Weg



<https://azure.microsoft.com/en-us/blog/microservices-an-application-revolution-powered-by-the-cloud/>



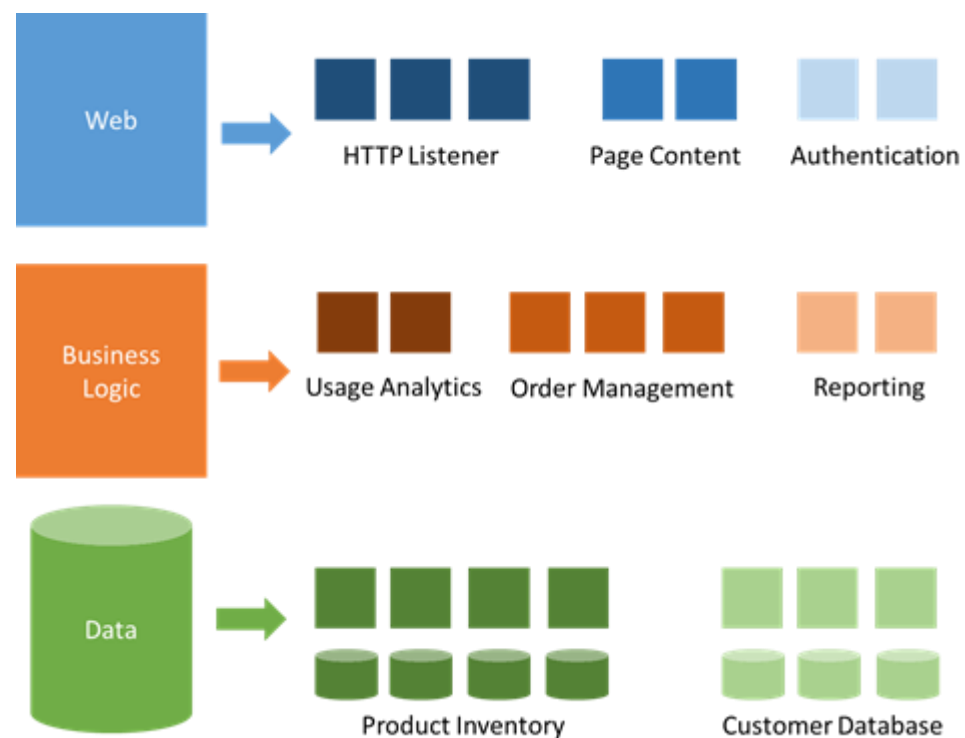
Der Microsoft - Weg



<https://azure.microsoft.com/en-us/blog/microservices-an-application-revolution-powered-by-the-cloud/>



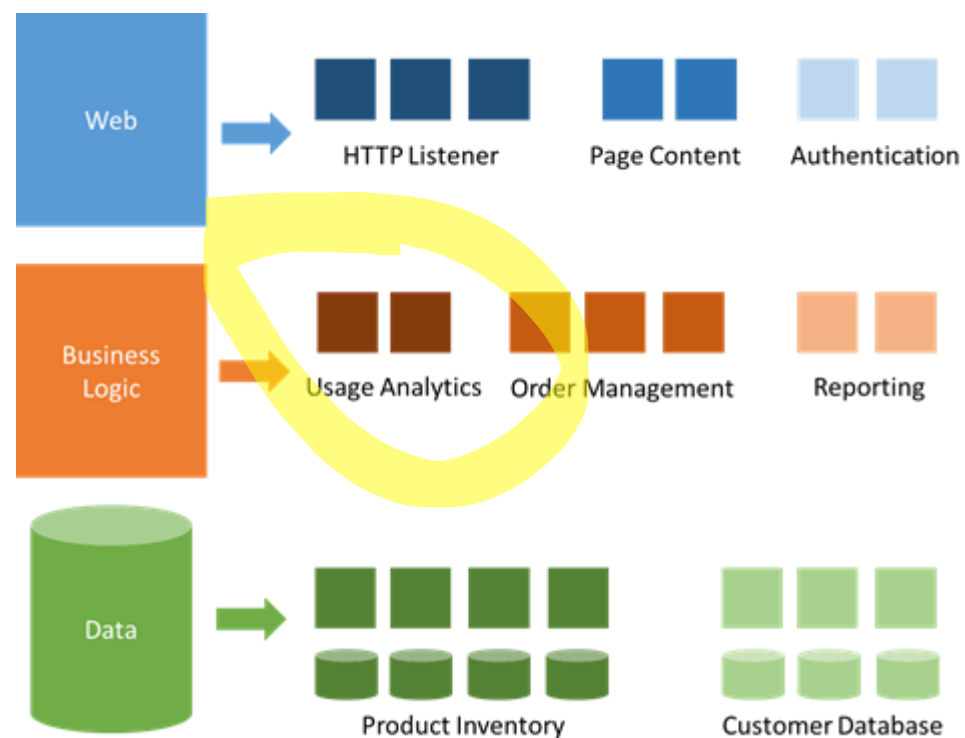
Der Microsoft - Weg



<https://azure.microsoft.com/en-us/blog/microservices-an-application-revolution-powered-by-the-cloud/>



Der Microsoft - Weg

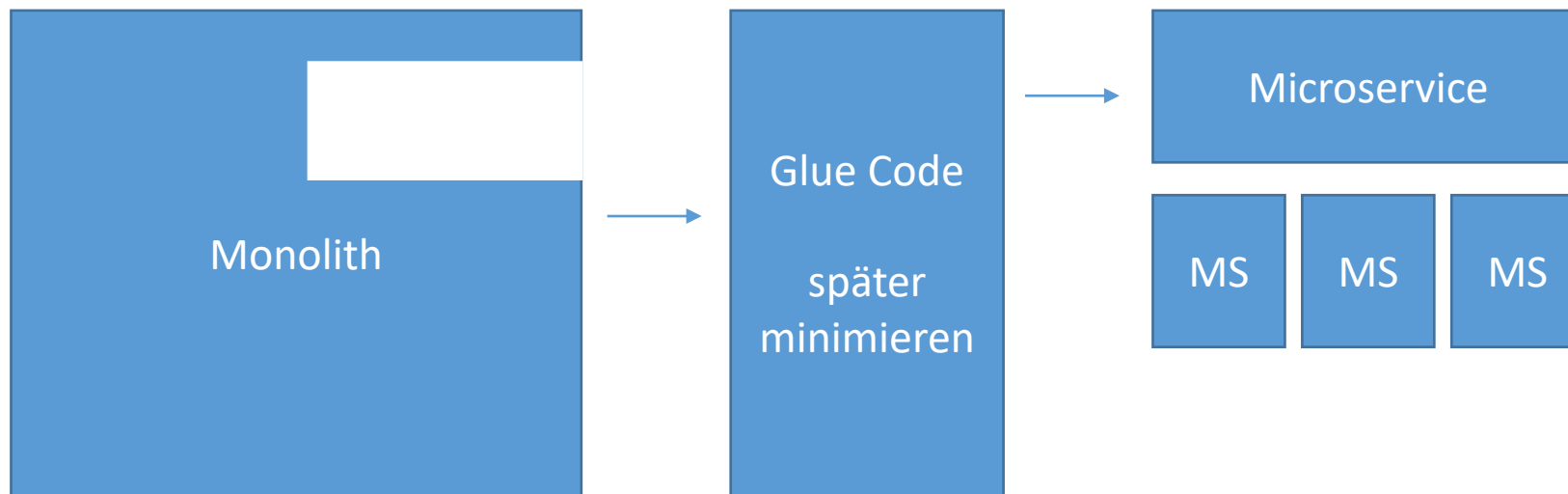


<https://azure.microsoft.com/en-us/blog/microservices-an-application-revolution-powered-by-the-cloud/>



Monolith First ist Pflicht

- Wer den Monolith nicht beherrscht, schafft die Microservices auch nicht
- Zielgerichtete Migration an den **relevanten** Punkten
 - „Gateway Migration“



Live Demo



NDepend



Goldener Monolith



Investment in den „goldenen Monolithen“

- Sich damit abfinden
- Wenn Monolith, aber dann richtig
- Alle unnötigen Architekturansätze raus werfen
- Abstraktionen raus
- Interne Modularisierung
 - Warum muss es ein externer Service werden?
 - „Internal Microservices“
- Neues UI
 - Blazor
 - WISEJ
 - App
 - ...

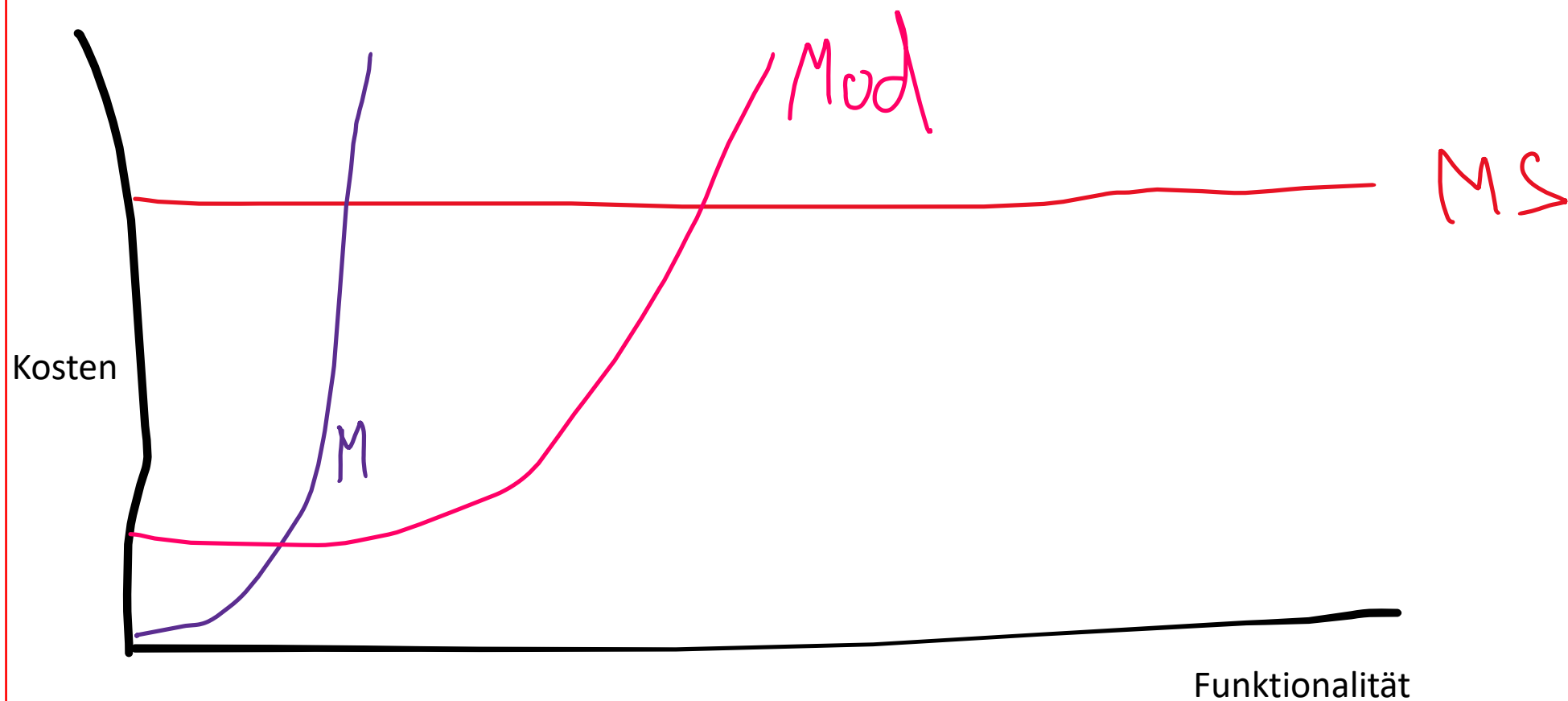


Investment in den „goldenen Monolithen“

- Klarheit schaffen
- Dieser Monolith wird niemals:
 - Wiederverwendet werden
 - So „für immer“ bestehen
 - Kann nur besser werden
 - „Innere Komplexität“ minimieren



Zusammenfassung



Ihre Erfahrungen?



Fragen?



Vielen Dank!

Stephan Hüwe
sh@huestel.de

<https://www.stephan-huewe.de/>
<https://www.linkedin.com/in/stephan-huewe-0983532/>

Photos: shutterstock



Weiterlesen

- <https://jamesbrucecodes.wordpress.com/2020/05/12/you-are-not-netflix-microservice-madness/>
- <https://m.signalvnoise.com/the-majestic-monolith/>
- <https://atechnologistspov.com/i-just-heard-that-monoliths-are-the-future-of-software-development-2190bf7f3c40>
- <https://martinfowler.com/bliki/MonolithFirst.html>
- <https://eng.uber.com/microservice-architecture/>
- <https://rubygarage.org/blog/monolith-soa-microservices-serverless>
- <https://samnewman.io/blog/2015/04/07/microservices-for-greenfield/>

