



Gut das ist?

Die umgekehrte Architekturbewertung eines Internet-Giganten

STEFAN ZÖRNER, EMBARC

microXchg 2015
Berlin, 12. Februar 2015

microxchg²⁰¹⁵

Gut das ist?

Die umgekehrte Architekturbewertung eines Internet-Giganten

Netflix - das größte Internet-Business in den USA - zeichnet sich zeitweise für ein Drittel des gesamten Downstream-Traffics des Webs verantwortlich. Die Erfolge der Video-on-Demand Plattform basieren nicht nur auf einer guten Geschäftsidee, sondern auch auf top-modernen, effizienten und robusten Technologien, Frameworks und Architekturansätzen. Was können wir daraus lernen? Ist es an der Zeit unsere Systeme und Architekturen in Microservices zu refactoren, große Datenbanksysteme aufzubrechen, polyglott zu programmieren und reaktive Ansätze zu verwenden?

Diese Session hilft Ihnen bei der individuellen Beantwortung dieser Fragen. In einer umgekehrten Architekturbewertung haben wir jene Anforderungen und Rahmenbedingungen herausgearbeitet, die man haben müsste, um die Netflix-Architektur als ideal zu bewerten. Welche Qualitätsaussagen müssten Ihnen wichtig sein? Zu welchen Kompromissaussagen müssten Sie „ja“ sagen? Welche Risiken müssten Sie eingehen und welche Rahmenbedingungen bräuchten Sie?



Umgekehrte Architekturbewertung

embarc.de

2

Stefan Zörner

- Softwareentwickler + -architekt bei embark in Hamburg
- Vorher oose, IBM, Mummert + Partner, Bayer AG, ...

Schwerpunkte:

- Softwarearchitektur (Entwurf, Bewertung, Dokumentation)
- Java Technologien



Stefan.Zoerner@embarc.de



[@StefanZoerner](https://twitter.com/StefanZoerner)



xing.to/szr



Umgekehrte Architekturbewertung

embarc.de

3

Agenda

- 1 Einstieg
- 2 Was ist Netflix?
- 3 Lösung im Detail
- 4 Konsequenzen
- 5 Weitere Informationen



Umgekehrte Architekturbewertung

embarc.de

4

Agenda

- 1 Einstieg
- 2 Was ist Netflix?
- 3 Lösung im Detail
- 4 Ergebnisse zusammengefasst
- 5 Weitere Informationen

1

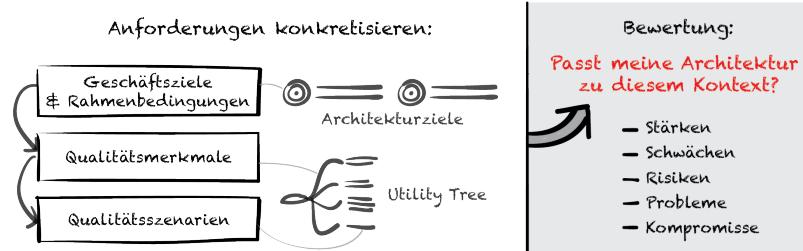


Umgekehrte Architekturbewertung

embarc.de

5

Architekturbewertung



Methoden:

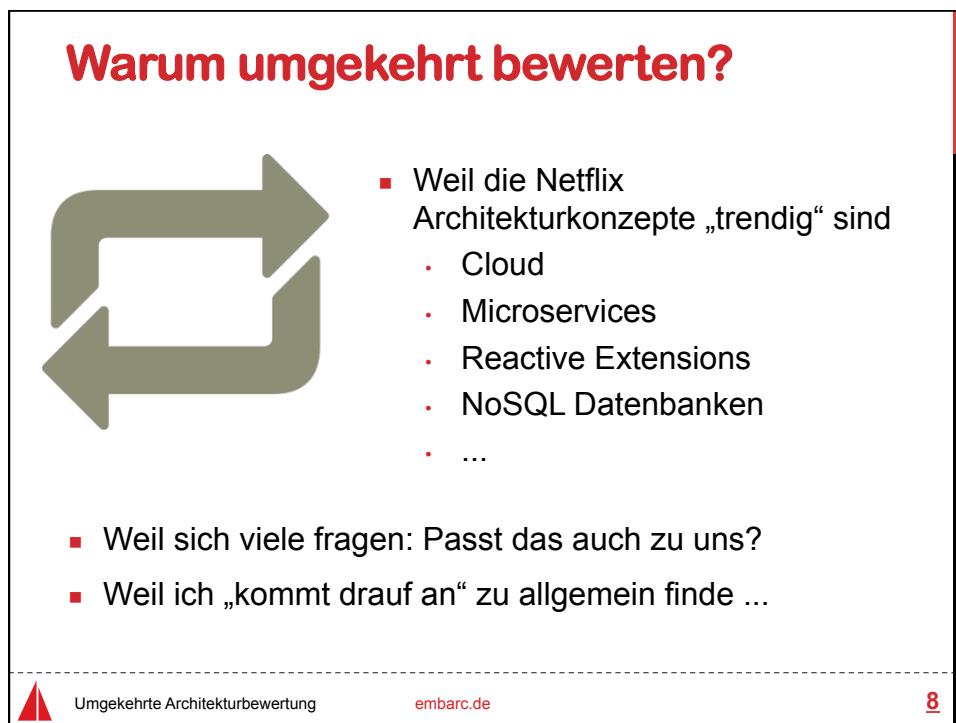
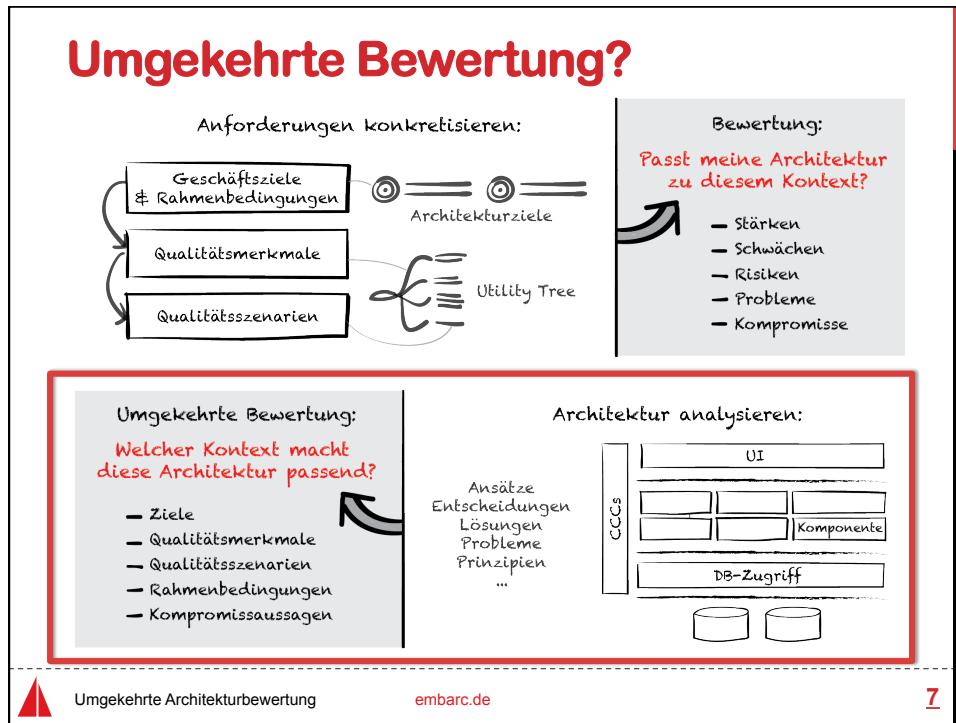
- **ATAM** – Architecture Tradeoff Analysis Method
- **CBAM** – Cost-Benefit Analysis Method
- **SACAM** – Software Architecture Comparison Analysis Method
- **LAAAM** – Lightweight Architecture Alternative Assessment Method
- ...



Umgekehrte Architekturbewertung

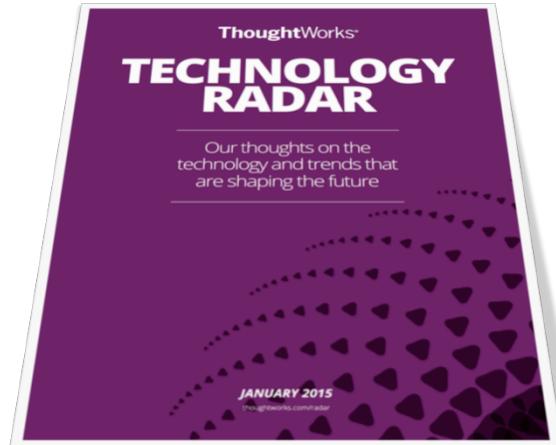
embarc.de

6



ThoughtWorks Technology Radar

→ <http://www.thoughtworks.com/radar/>



Umgekehrte Architekturbewertung

embarc.de

9

Agenda

2

- 1 Einstieg
- 2 Was ist Netflix?
- 3 Lösung im Detail
- 4 Ergebnisse zusammengefasst
- 5 Weitere Informationen



Umgekehrte Architekturbewertung

embarc.de

10

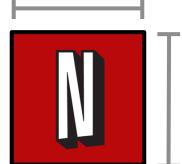


Was ist Netflix?

"Netflix is the world's leading Internet television network with over 57 million members in nearly 50 countries enjoying more than two billion hours of TV shows and movies per month, including original series. For one low monthly price, Netflix members can watch as much as they want, anytime, anywhere, on nearly any Internet-connected screen. Members can play, pause and resume watching, all without commercials or commitments."

→ <http://ir.netflix.com>

Netflix – Wie groß ist ‘groß’?



- 100+ Midtier-Services und Applikationen
- Milliarden Requests am Tag
- ~70 Milliarden Events am Tag
- 10.000s Ec2 Instanzen in mehreren AWS Regionen/Zonen
- Cassandra NoSQL in einem Regionen-übergreifenden, globalen Ring mit **Terabytes** an Daten
- Zu Spitzenzeiten **1/3** der Internet-Bandbreite in den USA



Umgekehrte Architekturbewertung

embarc.de

13

,Netflix is the king of online streaming, using more global bandwidth than cat videos and piracy combined.'

House of Cards
★★★★★ 2013 TV-MA 1 Season HD 5.1
Sharks gliding ominously beneath the surface of the water? They're a lot less menacing than this Congressman.
This winner of three Emmys, including Outstanding Directing for David Fincher, stars Kevin Spacey and Robin Wright.

Because you watched Orange Is the New Black
A NETFLIX ORIGINAL SERIES
HOUSE of CARDS
the L word
the League
ALWAYS SUNNY in PHILADELPHIA
New Girl

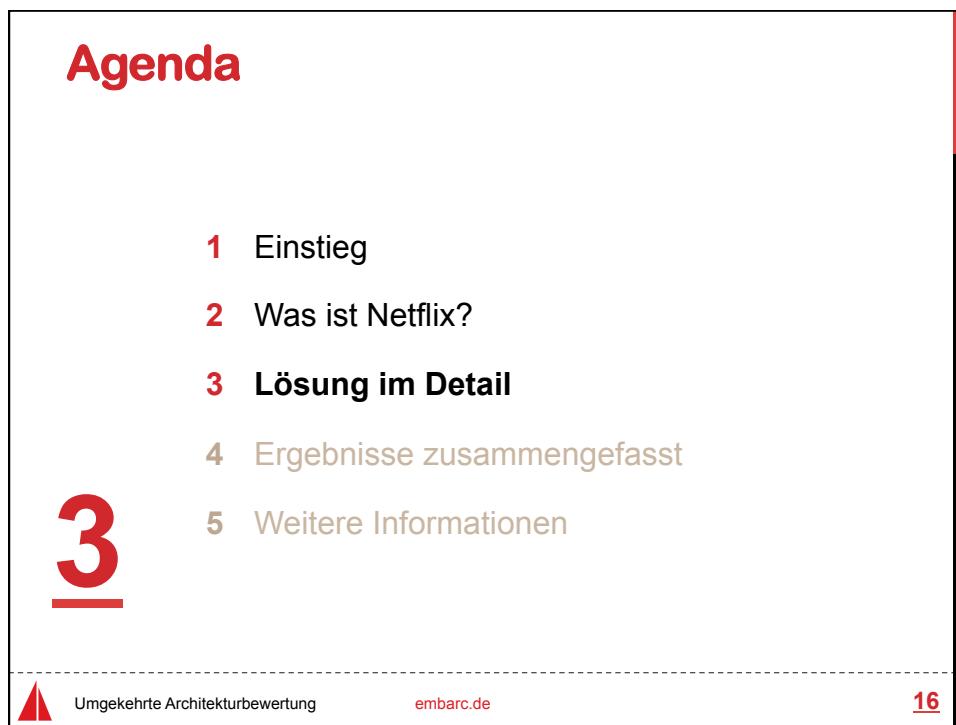
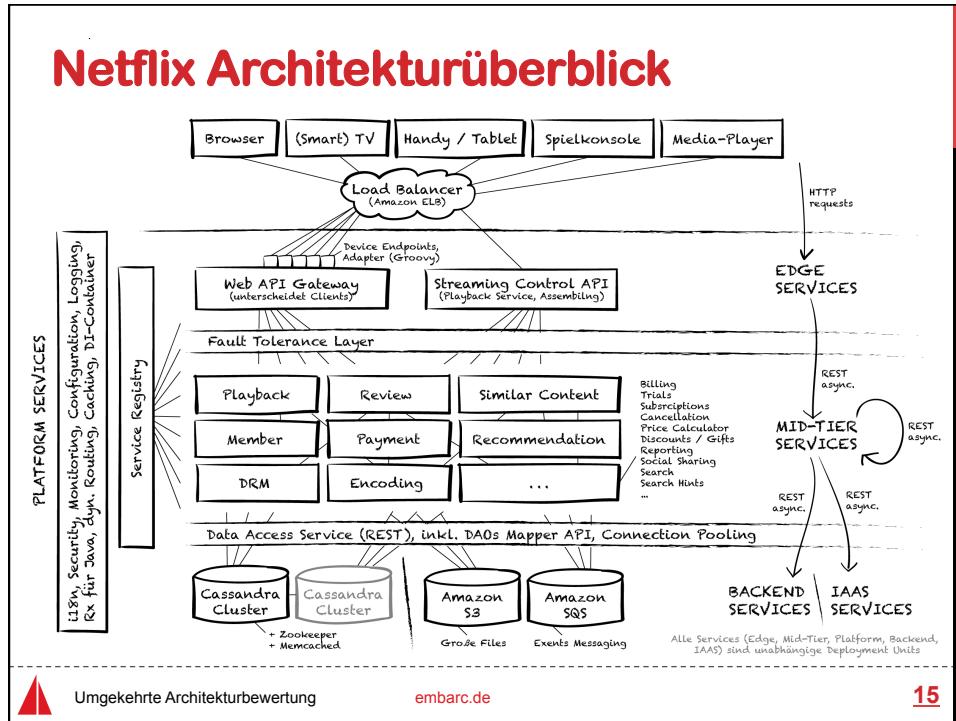
Because you watched Red Lights
6 SOULS
THE ELECTRIC MIST



Umgekehrte Architekturbewertung

embarc.de

14



Lösung im Detail (1)



Umgekehrte Architekturbewertung

embarc.de

17

Was sind Microservices?

"In short, the microservice architectural style is an approach to developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms, often an HTTP resource API."



(James Lewis, Martin Fowler)

Charakteristische Eigenschaften

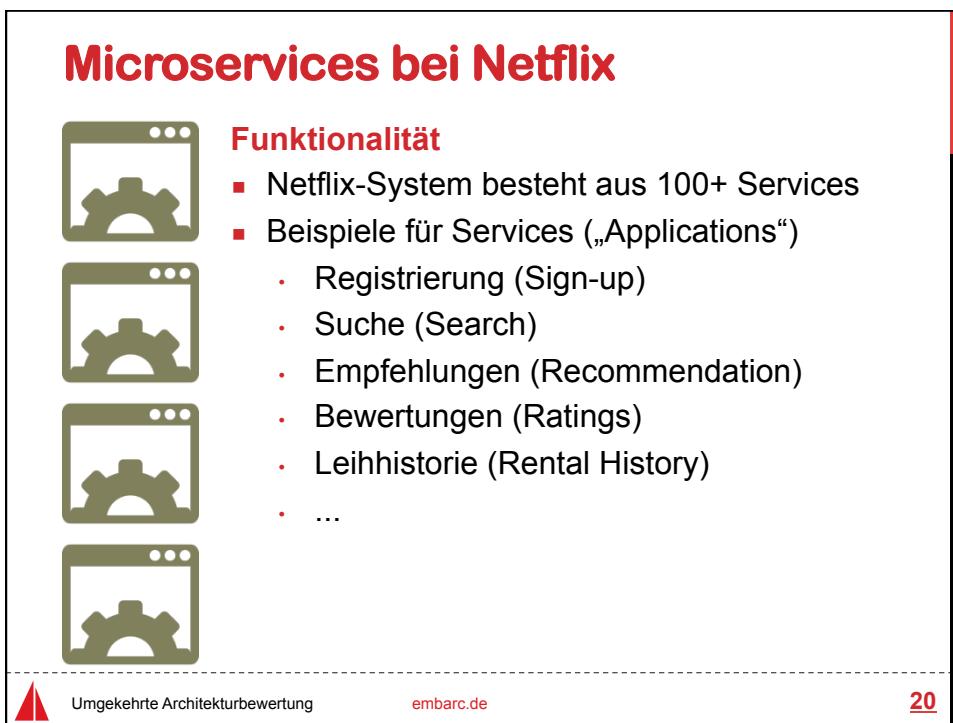
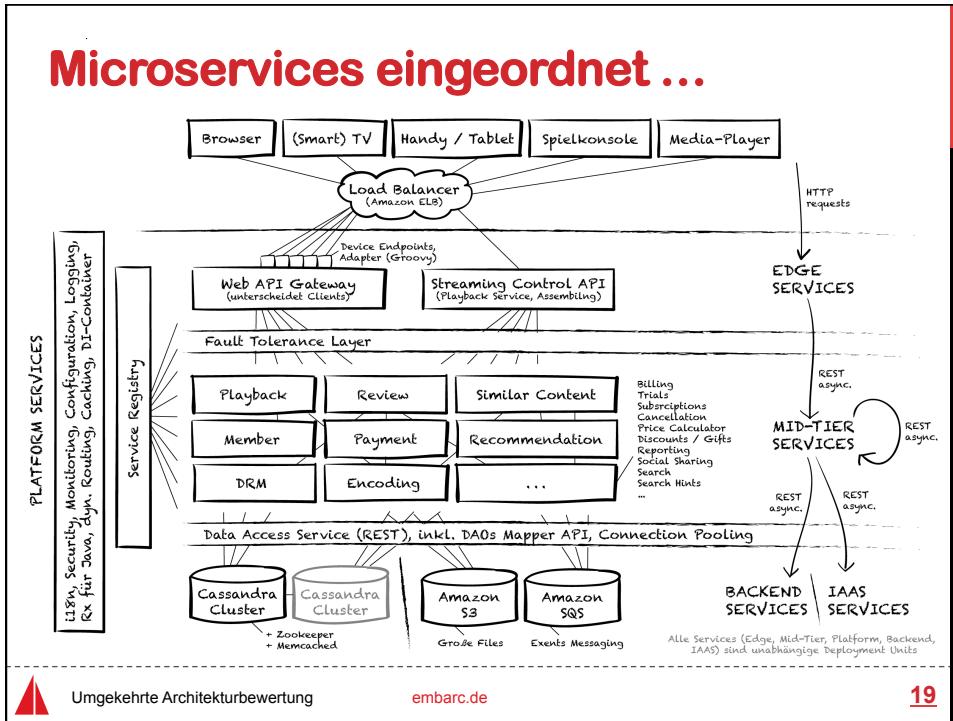
- Zerlegung in relativ kleine (fachliche) Services
- Services sehr lose gekoppelt
- Services einzeln installierbar und upgradebar
- Dezentrale Datenhaltung
- Hoher Freiheitsgrad bei Technologieauswahl



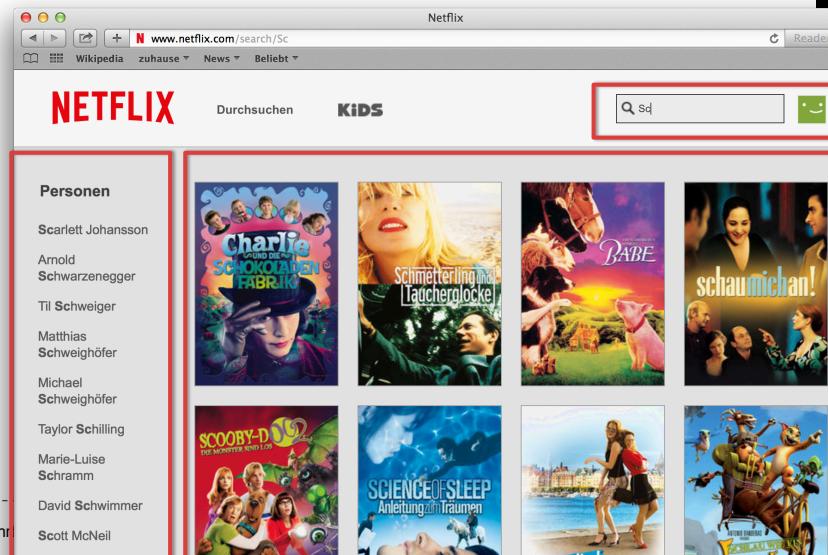
Umgekehrte Architekturbewertung

embarc.de

18



Beispiel für eine Service-Trennung



Microservices bei Netflix (org.)

Teams vollumfänglich für ihre Services **verantwortlich**

- Entwicklung
- Release und Deployment
- Betrieb

Keine klassische **Management-Steuerung**

Keine Beeinflussung durch andere Teams oder
eine zentrale Instanz

- Wenige technologische Vorgaben
- Release nach Belieben

“Freedom & Responsibility”



Verwendete Technologien

Plattformen		Persistenz
Apache HTTP Server		Cassandra
Apache Tomcat		RDBMS (MySQL)
Bottle (Python)		in-memory caches
...		Amazon S3

Programmiersprachen	
Java	Groovy
Scala	Python
JavaScript	Clojure
Dart	Ruby
C++	...

Umgekehrte Architekturbewertung embarc.de 23

Konsequenzen für Netflix

- Neue Technologien sind leicht ausprobiert
- Technologie-Stack kann nach und nach modernisiert werden (keine langfristige Binding)
- Fehler in einer Applikation haben wenig Einfluss auf andere Applikationen
- Time-to-Market für neue Funktionalität



Herausforderungen

- Die Gesamtmenge der verwendeten Technologien ist sehr heterogen
- Abstimmung / Koordination schwierig
- Jedes Service-Team muss Know-How für eigentlich querschnittliche Themen haben

Umgekehrte Architekturbewertung embarc.de 24

Lösung im Detail (2)

Netflix OSS Komponenten



Umgekehrte Architekturbewertung

embarc.de

25

Netflix Cloud Stack

Individuelle Software

Applikationen, Services, ...

PaaS

Laufzeitumgebung, Web-/Application-Server, Frameworks
für querschnittliche Aspekte, Management Tools, ...

NETFLIX | OSS

IaaS

Virtuelle Maschinen, Netzwerkkommunikation,
Load Balancing, Datenspeicher, ...



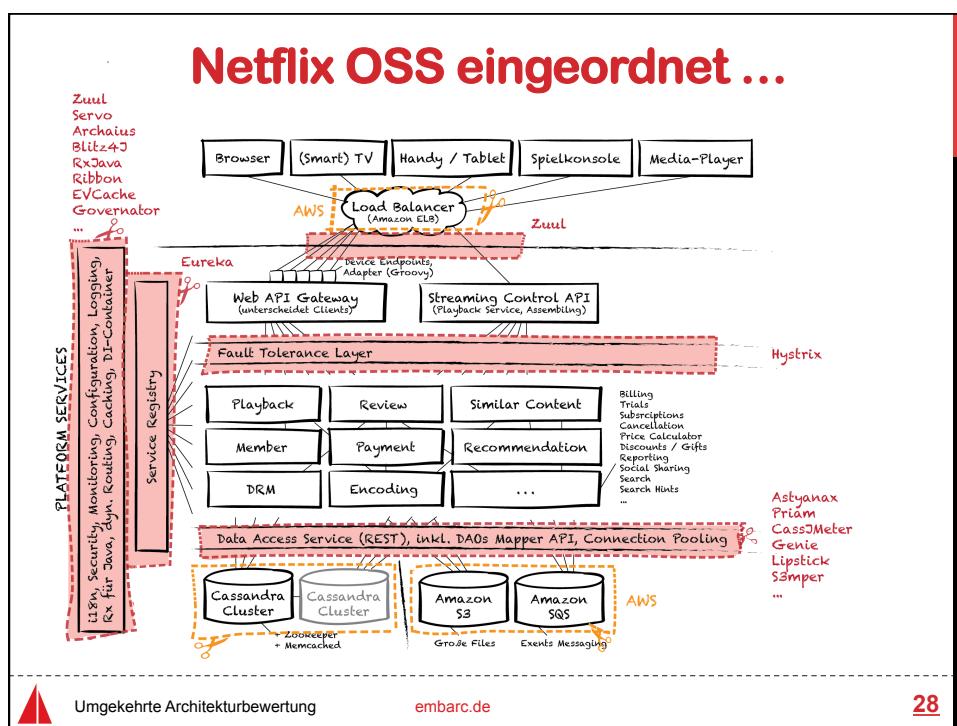
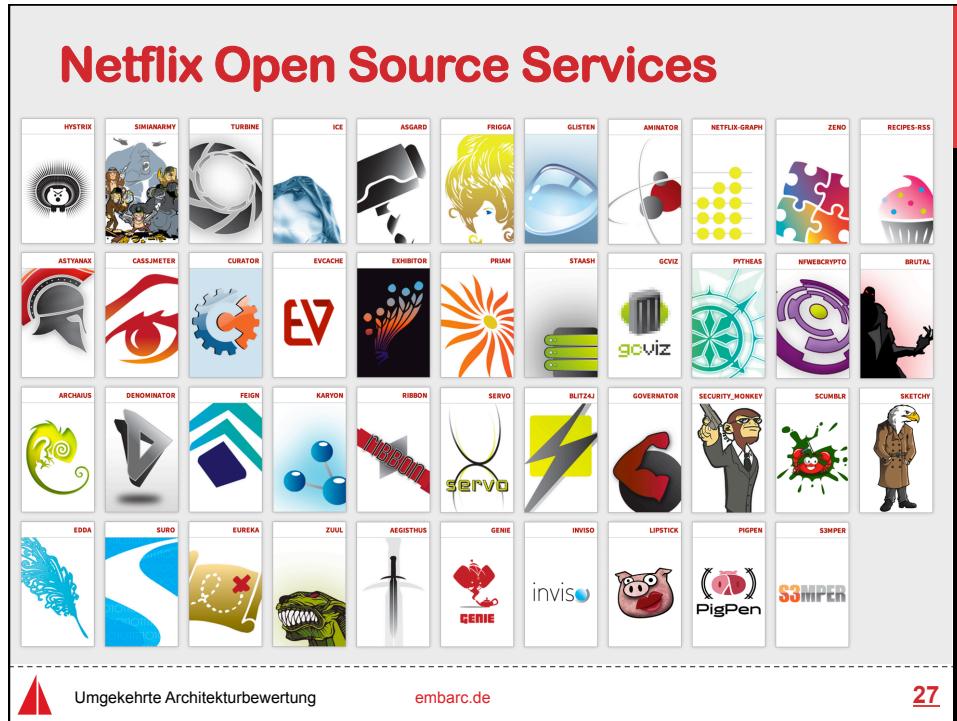
The Netflix Open Source Platform Components fill gaps in Amazon Web Services. The goal is to make cloud infrastructure more robust, flexible and glitch free.



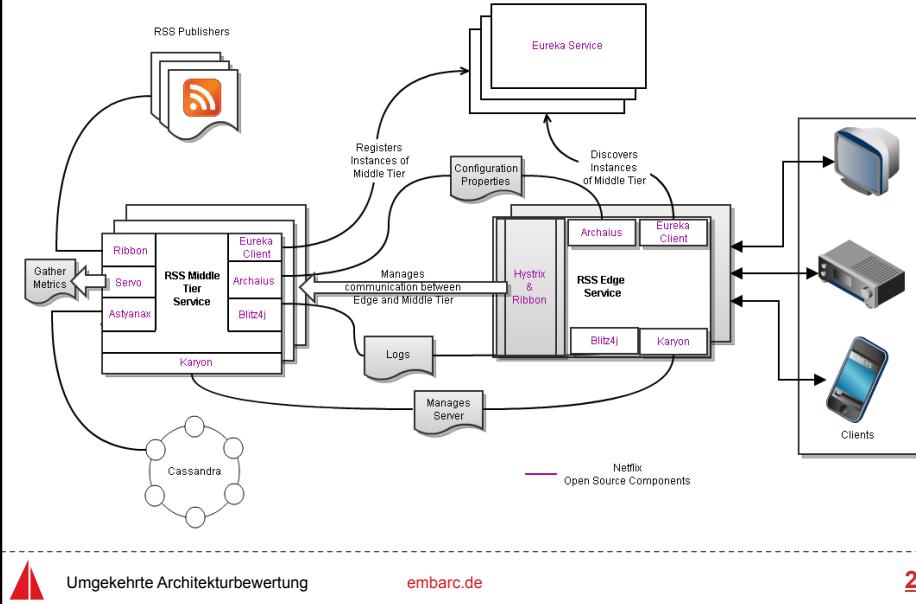
Umgekehrte Architekturbewertung

embarc.de

26



Beispielanwendung (2 Services)



Konsequenzen für Netflix

- Anforderungen an Entwickler sinken
- Schnelle Time-to-Market für neue Features
- Architektur-Prinzipien werden unterstützt (Serviceaufbau, Kommunikation, Verwendung von 3rd Party etc.)
- „Open-sourcing“ von Projekten fördert:
 - Ein Mindestmaß an Dokumentation (Wiki, Techblog)
 - Sauberes Design
 - Bild als interessanter Arbeitgeber



Herausforderungen

- Netflix-Spezifika in Entwicklung recht prominent
- Neues Projekt auf dieser Basis hat Overhead

Lösung im Detail (3)

Netflix Deployment



Umgekehrte Architekturbewertung

embarc.de

31

Deployment bei Netflix

Antwort auf **Koordinationsproblem** bei Deployments?

Antwort auf **Komplexität** bei Teamabhängigkeiten?

Unterstützte Anarchie

- **ca. 100** Deployments pro Tag
- Teams arbeiten selbstverantwortlich und unabhängig
- **Keine** Qualitätssicherungsabteilung
- **Keine** Release Engineers
- **Keine** Gesamtkoordination von Releases / Deployments

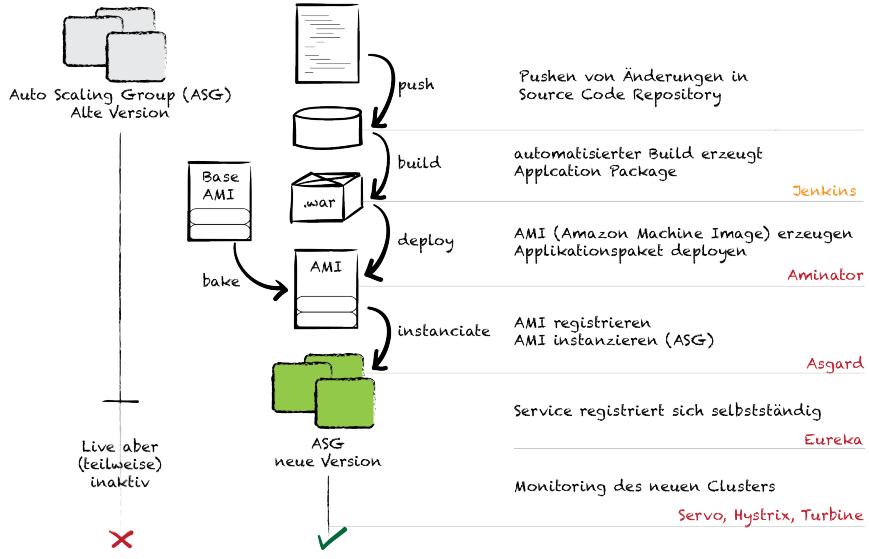


Umgekehrte Architekturbewertung

embarc.de

32

Komplette Automatisierung ...



Konsequenzen für Netflix

- Schnelles Rollback
- Produktionsfehler, Abhängigkeitsprobleme schlagen nicht zum Kunden durch
- Toolkette abstrahiert und entkoppelt Entwickler von zentralen Einstellungen und Konfigurationen



Herausforderungen

- Doppelte Hardware erforderlich
- Koordinationsprobleme auf First-come-first-serve heruntergebrochen
- Hohe Anforderungen an Logging, Monitoring in Go-Live-Phase



Lösung im Detail (4)

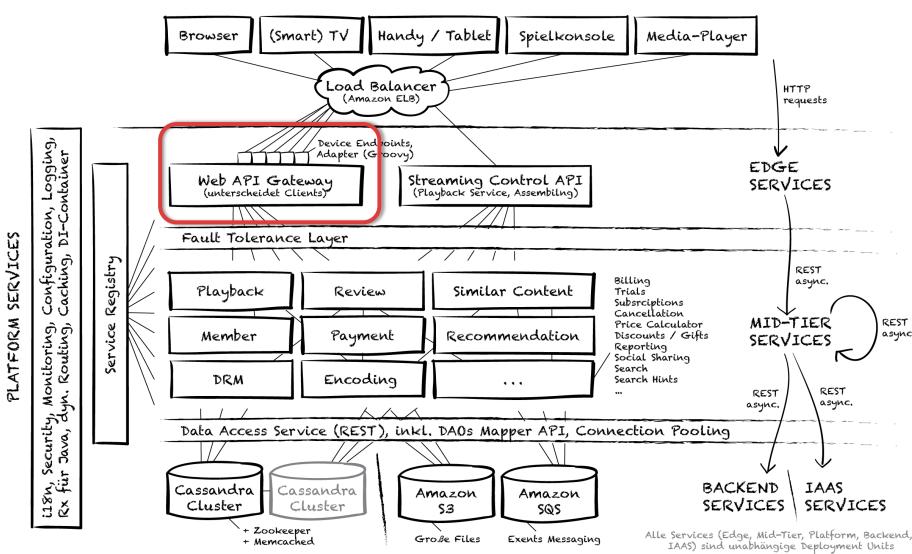


Umgekehrte Architekturbewertung

embarc.de

35

Netflix API eingeordnet ...

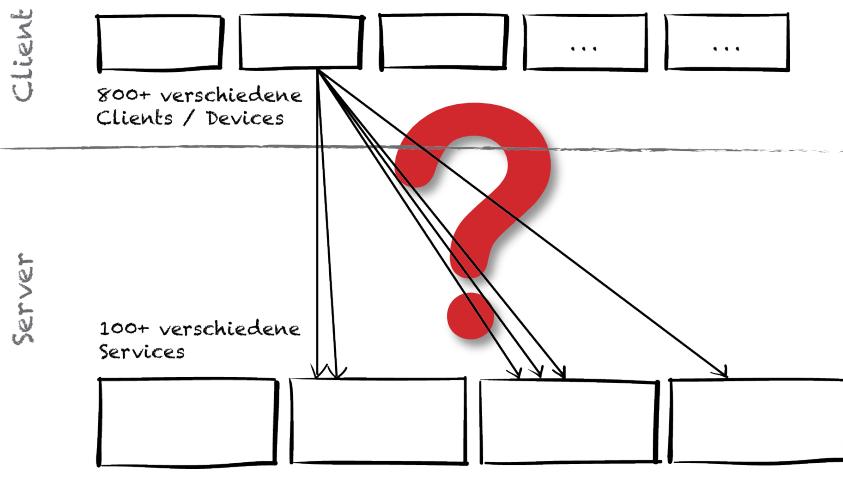


Umgekehrte Architekturbewertung

embarc.de

36

Wie nutzen / kombinieren Clients die Services?

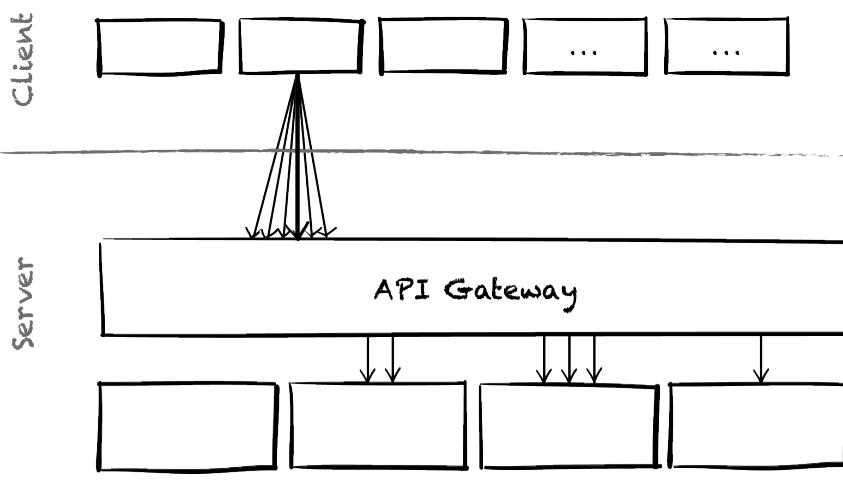


Umgekehrte Architekturbewertung

embarc.de

37

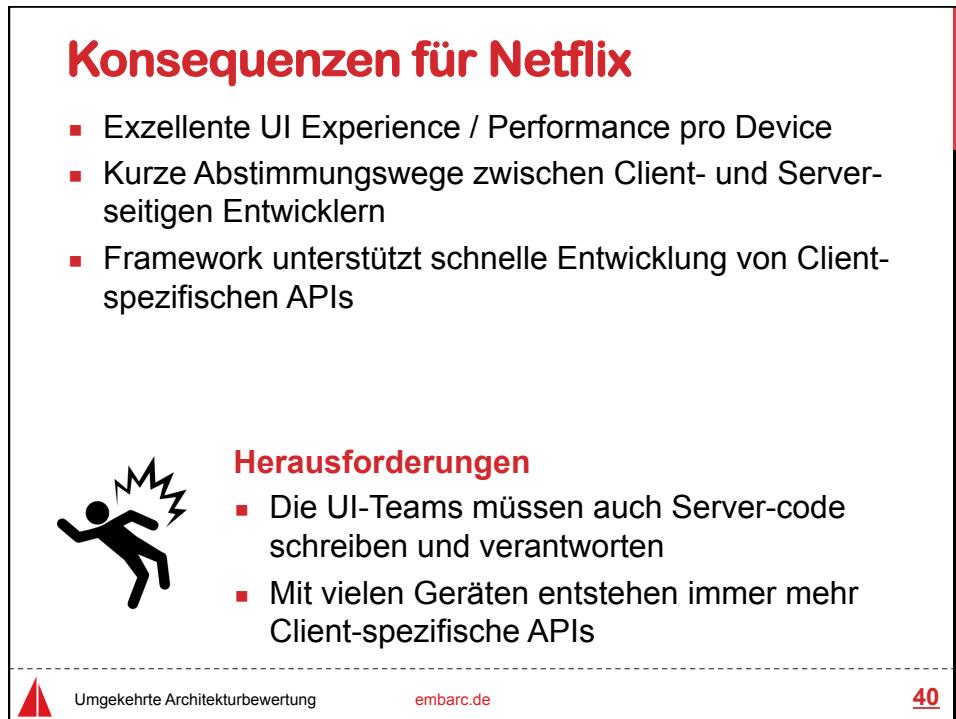
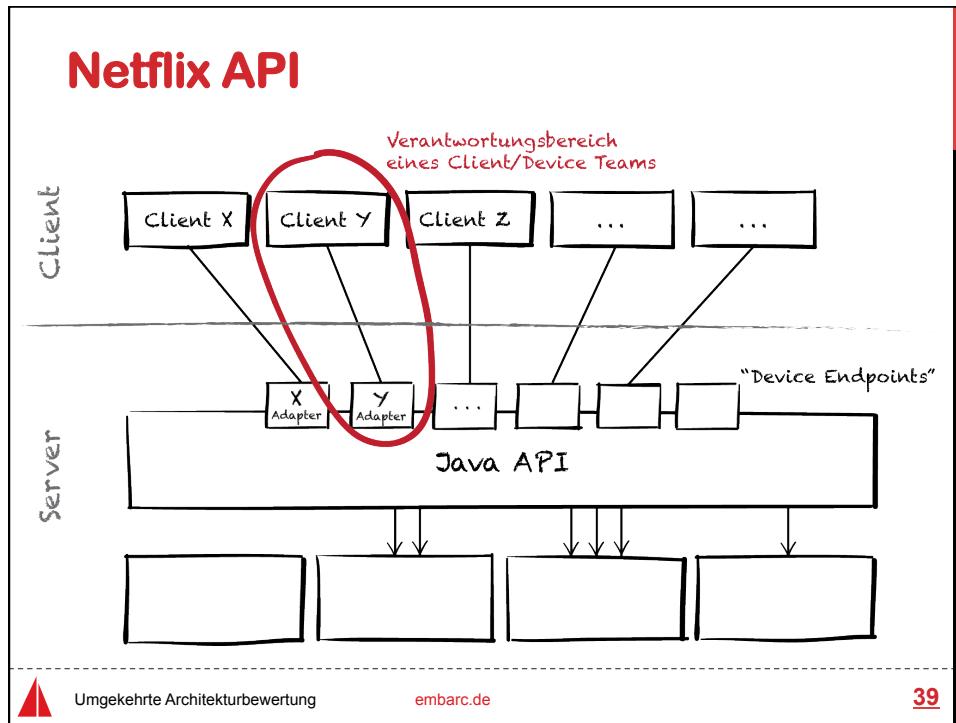
API Gateway („Remote Facade“)



Umgekehrte Architekturbewertung

embarc.de

38



Agenda

- 1 Einstieg**
- 2 Was ist Netflix?**
- 3 Lösung im Detail**
- 4 Ergebnisse zusammengefasst**
- 5 Weitere Informationen**

4



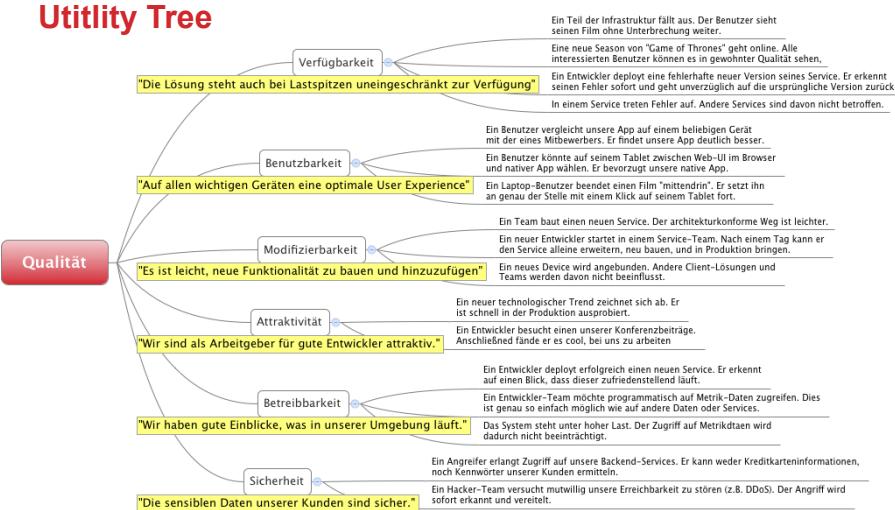
Umgekehrte Architekturbewertung

embarc.de

41

Qualitätsanforderungen

Utility Tree



Umgekehrte Architekturbewertung

embarc.de

42

Zentrale Randbedingungen

Welche Rahmenbedingungen bräuchte man aus unserer Sicht für die Netflix-Lösung?



1. (Weiter-)Entwicklung eines **langlebigen Produktes**
2. Größe des Produktes rechtfertigt **mehrere Teams**
3. Teams können **selbstbestimmt** arbeiten
4. Deployment in **Cloud** machbar
5. Scheitern im Deployment möglich
6. Einsatz von **Open Source-Lösungen** OK



Zusammenfassende Kompromissaussagen

NETFLIX



Technologieentscheidungen auf Teamebene und Experimente zur besseren Erreichung qualitativer Ziele sind einer homogenen Landschaft mit hoher Integrität vorzuziehen.

NETFLIX



Umgekehrte Architekturbewertung

embarc.de

45

Die schnelle Erstellung und Auslieferung von Funktionalität ist wichtiger als gut getestete Lösungen und „Fehlerfreiheit“ in Produktion.

NETFLIX



Umgekehrte Architekturbewertung

embarc.de

46

Innovationskraft und Wachstum sind sehr wichtige Aspekte der Softwareentwicklung.

Kontrolle, Planungsfähigkeit und Statusaussagen gegenüber dem Management sind diesen Aspekten klar untergeordnet.

NETFLIX



Umgekehrte Architekturbewertung

embarc.de

47

Redundante Entwicklungstätigkeiten und geringere Wiederverwendungs möglichkeiten sind jederzeit in Ordnung, um herausragende Qualität und entsprechende Marktvorteile zu erzielen.

NETFLIX



Umgekehrte Architekturbewertung

embarc.de

48

Es ist wichtig auf schwankende Systemlast, sowie Hard- und Software-Fehler zu reagieren ohne Verfügbarkeit einzubüßen. Kosten für Redundanzen und etwaige Inkonsistenzen sind geringe Opfer.

NETFLIX



Umgekehrte Architekturbewertung

embarc.de

49

Die langfristige Brauchbarkeit und zeitgemäße Ausprägung der Lösung rechtfertigt höhere initiale Aufwände für Framework- und Infrastrukturkomponenten.

NETFLIX



Umgekehrte Architekturbewertung

embarc.de

50

Agenda

- 1 Einstieg
- 2 Was ist Netflix?
- 3 Lösung im Detail
- 4 Ergebnisse zusammengefasst
- 5 Weitere Informationen

5



Umgekehrte Architekturbewertung

embarc.de

51

Netflix Tech Blog

→ <http://techblog.netflix.com>

The screenshot shows the Netflix Tech Blog homepage. The header features the Netflix logo and the text "The Netflix Tech Blog". Below the header, a news article is displayed with the title "Nicobar: Dynamic Scripting Library for Java". The article includes a small circular graphic with blue and green circles, a date ("Tuesday, February 10, 2015"), and author information ("By James Kojo, Vasanth Asokan, George Campbell, Aaron Tull"). The main content discusses the Nicobar library, which allows UI teams to inject device-specific adapter code dynamically into their JVM without API involvement. A sidebar on the right is titled "Links" and lists various Netflix blogs and websites.

Software Engineering Radio #216

→ <http://www.se-radio.net/2014/12/episode-216-adrian-cockcroft-on-the-modern-cloud-based-platform/>

The screenshot shows a web browser displaying the Software Engineering Radio website. The main title is "Software Engineering Radio" with the subtitle "The Podcast for Professional Software Developers". Below the header is a navigation bar with links to Home, About SE-Radio, Be A Host, Praise, Team, Contact, and Legal. The main content area features a large image of Adrian Cockcroft and the title "Episode 216: Adrian Cockcroft on the Modern Cloud-based Platform". A small note below the title indicates it was filed in "Episodes" by "SE-Radio" on December 10, 2014, with 7 comments. To the right of the main content is a "Connect" sidebar with social media icons for Facebook, Twitter, Google+, LinkedIn, and RSS. Another sidebar on the right displays a thumbnail for "IEEE Computer Society Software" and a red box containing the number "53".

Netflix Open Source Software

→ <http://netflix.github.io>

The screenshot shows a web browser displaying the Netflix Open Source Software Center website. The header features the Netflix OSS logo and the text "Netflix Open Source Software Center". Below the header is a navigation bar with "Repositories" and "Powered By NetflixOSS". The main content area is titled "Availability" and shows several projects: HYSTRIX, SIMIANARMY, TURBINE, ICE, ASGARD, FRIGGA, and GLISTEN. A detailed view of the SIMIANARMY project is shown in a modal window, which includes a description of the tool, statistics (Stars: 2445, Forks: 298, Language: Java, Open Issues: 26), and an "Updated" timestamp (02/11/15 @ 18:32:30). To the right of the modal is a sidebar with a message about maintaining subscriptions to mailing lists for various projects. At the bottom right of the sidebar is the number "54".

Vielen Dank.

Ich freue mich auf Eure Fragen!

✉ stefan.zoerner@embarc.de

🐦 @StefanZoerner

X xing.to/szr



DOWNLOAD FOLIEN:

<http://www.embarc.de/blog/>



Vielen Dank.

Ich freue mich auf Ihre Fragen!

✉ stefan.toth@embarc.de

🐦 @st_toth

X xing.to/sto



DOWNLOAD FOLIEN:

<http://www.embarc.de/blog/>