31.8.–3.9.2015
in Nürnberg

Herbstcampus

Wissenstransfer
par excellence

So sieht‘s aus!

Architekturüberblicke: Tipps und Tricks

Stefan Zörner
embarc GmbH, Hamburg

embarc
Software Consulting GmbH

So sieht‘s aus!

Architekturüberblicke: Tipps und Tricks

Ihr habt eine bestehende Softwarelösung und wollt deren Architektur erklären? Vielleicht einem neuen Teammitglied oder einem Kooperationspartner. Der Quelltext ist zu „ausführlich“ und erzählt trotzdem nicht die ganze Geschichte. Was zeigt Ihr in solchen Fällen von Eurer Dokumentation? Ach habt Ihr nicht …
Nicht schlimm: In diesem Vortrag geht es um das Anfertigen eines Architekturüberblicks aus dem Nichts heraus. Das stumpfe Ausfüllen eines Templates führt dabei oft zu Frust. Ich zeige stattdessen, wie Ihr die Aufgabe pragmatisch und zugleich wirkungsvoll angeht. Dabei diskutieren wir auch Formen, Zutaten, Werkzeuge und Vorgehen mit an. Ihr lernt die minimale Ausprägung eines Architekturüberblicks kennen und erfahrt, in welchen Situationen Ihr von diesem Minimalset abweichen könnt.

Stefan Zörner

- Softwareentwickler + -architekt bei embarc in Hamburg
- Vorher oose, IBM, Mummert + Partner, Bayer AG, …

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

Stefan.Zoerner@embarc.de
@StefanZoerner
xing.to/szr

embarc Software Consulting GmbH

Agenda

1. Motivation
2. Formen und Zutaten
3. Zutaten der Problemstellung
4. Zutaten der Lösung
5. Die Zubereitung
6. Fazit und weitere Informationen.

Agenda

1 **Motivation**

2 Formen und Zutaten

3 Zutaten der Problemstellung

4 Zutaten der Lösung

5 Die Zubereitung

6 Fazit und weitere Informationen.

1

Hampton Court Palace

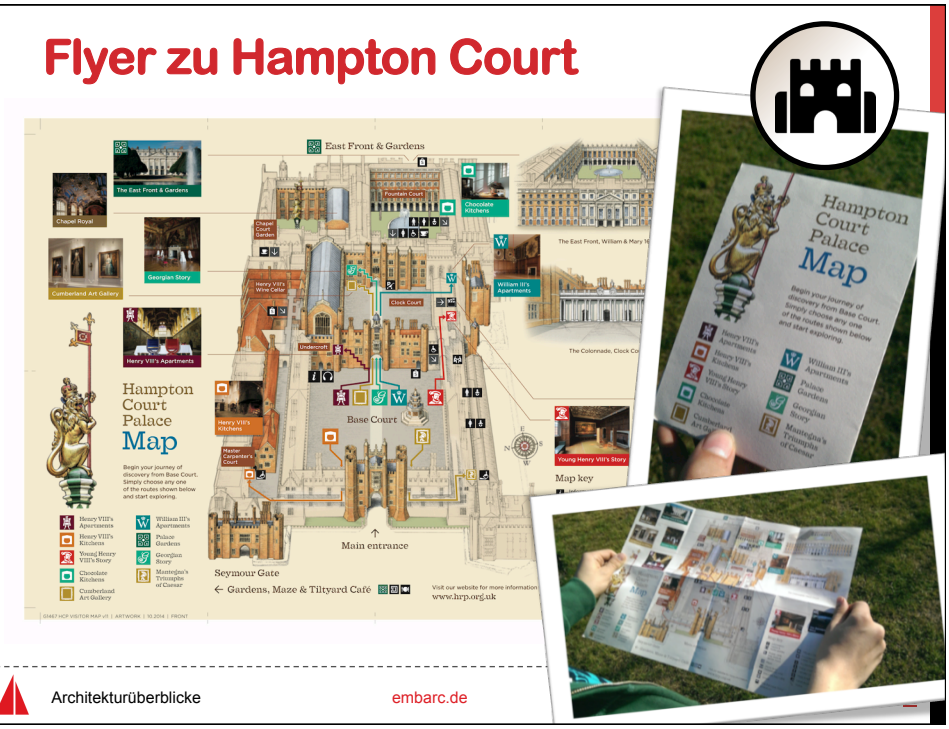
Schloss im äußersten Südwesten Londons
Residenz Heinrichs VIII. (Elisabeth I., …)

Einige Kennzahlen und Features:

- erbaut 1515 (mehrmals erweitert und umgebaut)
- Anzahl Architekturstile: 2 (Tudor, Barock)
- Anzahl Bewohner (Hofstaat 1546): 1.300
- Größe der Anlage: 2,43 ha
- Park mit berühmtem Heckenlabyrinth („Maze“)
- Anzahl Schlossgespenster: > 3

→ http://www.hrp.org.uk/HamptonCourtPalace/

Flyer zu Hampton Court



«Euer Softwaresystem»

Anwendung im Zentrum Eurer Systemlandschaft
Residenz essenzieller Geschäftsprozesse

Einige Kennzahlen und Features:

- erbaut ca. 1998 (mehrmals erweitert und umgebaut)
- Anzahl Architekturstile: N > 2 (Schichten, SOA, ...)
- Größe der Anlage: > 400.000 LoC (> 1.000 Klassen)
- Anzahl Entwickler: 10+ (keiner von Anfang an)
- viele eingesetzte Frameworks, Bibliotheken ...
- Module mit berühmten Codelabyrinthen (Mazes)
- Anzahl Leichen im Keller: > 3

Flyer zu «Euer Softwaresystem»

(keiner)
Nicht vorhanden: Aktuelle, konsistente, angemessene Dokumentation

Worum geht's?

Eurer Team tut sich schwer, die Eckpunkte seiner Software-Lösung zu kommunizieren.

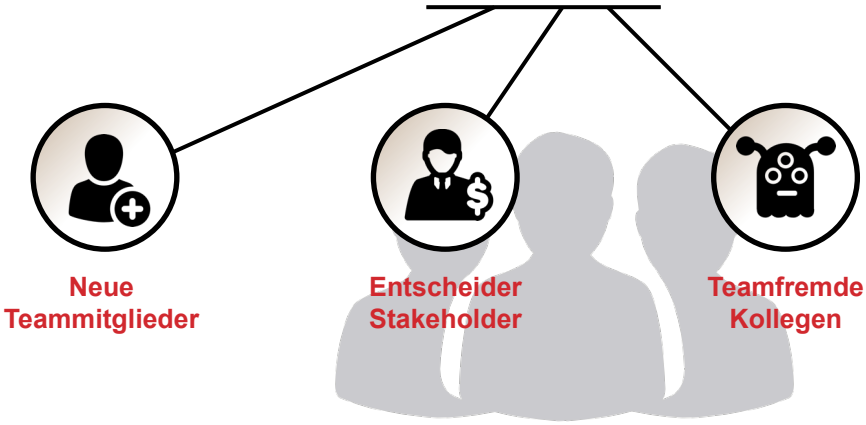
- **Neue Teammitglieder**, die mitentwickeln wollen, finden sich in der Architektur nicht zurecht.

- **Entscheider** und andere Stakeholder haben Unsicherheiten oder geringes Vertrauen in Eure Lösung.

- **Teamfremde Kollegen** sind an Lösungsansätzen interessiert, finden aber keine oder sehr detaillierte Informationen.

Was ist ein Architekturüberblick?

Ein Architekturüberblick macht die zentralen Lösungsansätze Eurer Softwarearchitektur für Außenstehende nachvollziehbar.



Agenda

2

1 Motivation
2 **Formen und Zutaten**
3 Zutaten der Problemstellung
4 Zutaten der Lösung
5 Die Zubereitung
6 Fazit und weitere Informationen.

Formen von Überblicken: Hampton Court

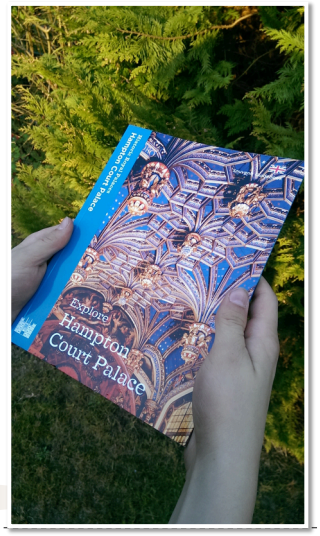
Webseite

Hinweisschilder

Formen von Überblicken: Hampton Court

Flyer

Broschüren



Formen eines Architekturüberblicks

Dokument

Foliensatz

Architekturportal (Wiki)

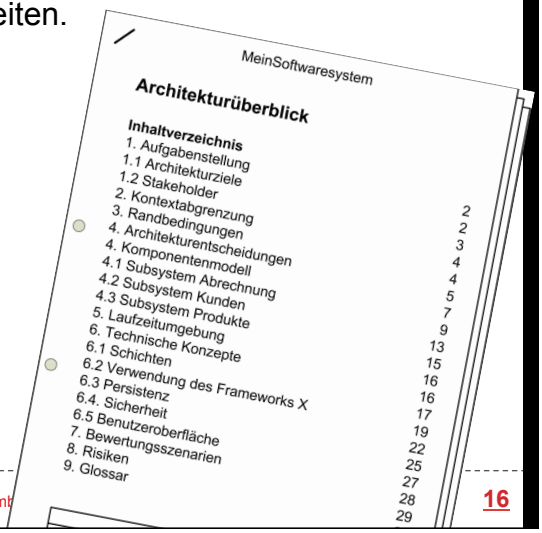
Wand oder Poster

Architektur-Flyer

Video

Prägnantes Dokument

Strukturierter Text, angereichert mit Illustrationen, Umfang maximal 30 Seiten.



MeinSoftwaresystem

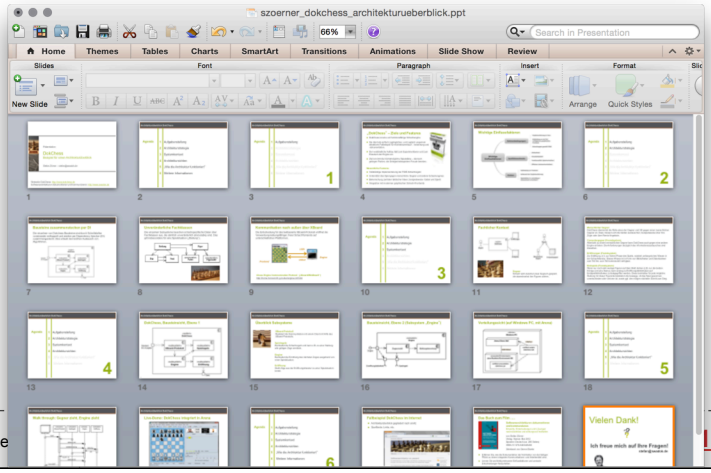
Architekturüberblick

Inhaltverzeichnis

1. Aufgabenstellung 2
1.1 Architekturziele 2
1.2 Stakeholder 3
2. Kontextabgrenzung 4
3. Randbedingungen 4
4. Architekturentscheidungen 5
4. Komponentenmodell 7
4.1 Subsystem Abrechnung 9
4.2 Subsystem Kunden 13
4.3 Subsystem Produkte 15
5. Laufzeitumgebung 16
6. Technische Konzepte 16
6.1 Schichten 17
6.2 Verwendung des Frameworks X 19
6.3 Persistenz 22
6.4. Sicherheit 25
6.5 Benutzeroberfläche 27
7. Bewertungsszenarien 28
8. Risiken 29
9. Glossar

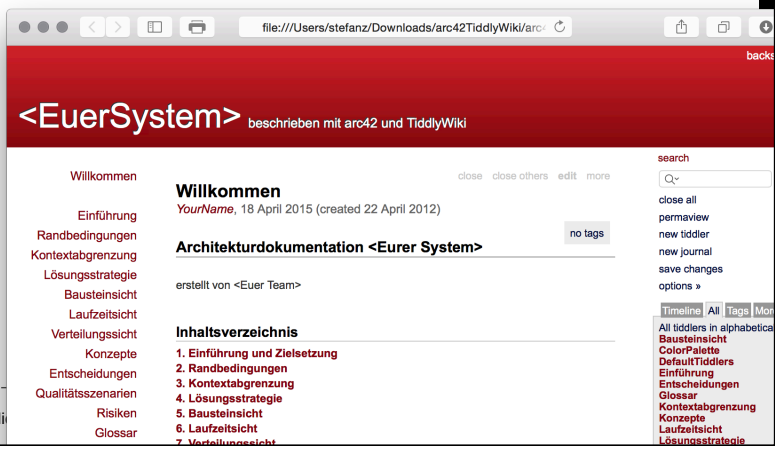
Foliensatz

10 - 15 Folien zur Unterstützung einer Präsentation der Architektur.



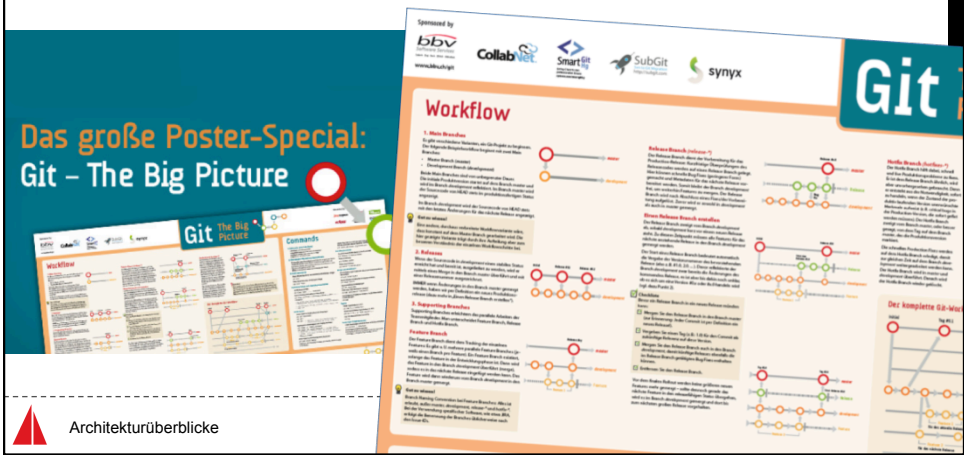
Architekturportal im Wiki

Einstiegseite(n) im Wiki, die Interessierte durch die Inhalte führen.



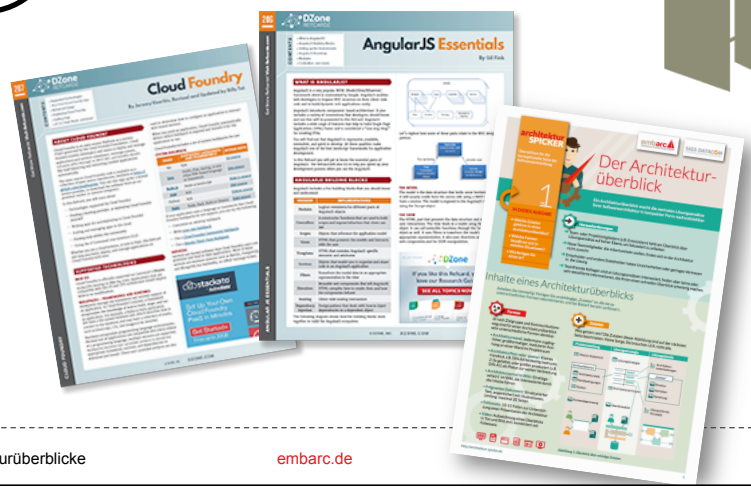
Architekturwand oder -poster

Jedermann zugängliche, großformatige Zusammenschau, modular/einmalig oder produziert als Poster zur weiten Verbreitung.



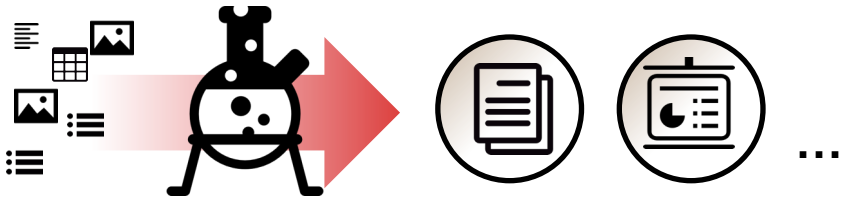
Architektur-Flyer

Kleines Handout, z.B. Din-A4 beidseitig bedruckt, 1 - 2x gefaltet.



Form(en) und Inhalt(e)

- Die Wahl der Form hängt u.a. ab von
  - Zielgruppe und Ziel des Überblicks
  - Zeitpunkt im Vorhaben/Projekt
- Die Formen schließen sich insbesondere nicht aus.
- Wenn Ihr kleinteilig arbeitet, habt Ihr schnell was Vorzeigbares ...



Zutaten in der Architekturdokumentation

Problemstellung

- Mission Statement
- Kontextabgrenzung
- Architekturziele
- Entscheidende Randbedingungen
- Größte Risiken

Lösungsstrategie

- Lösungsstrategie
- Architekturprinzipien
- Informelles Überblicksbild („Big Picture“)

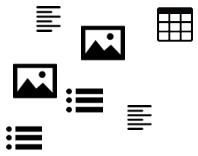
Lösungsdetails

- Architektur-entscheidungen
- Zerlegung
- Verhalten
- Verteilung
- Konzepte

Warum Zutaten?

Kleine, unabhängige Teile lassen sich ...

- schnell anfertigen
- schrittweise verfeinern
- leicht rekombinieren



Wichtig: Kein loses Schüttgut,
sondern Zusammenspiel mit rotem Faden

Agenda

1 Motivation

2 Formen und Zutaten

3 **Zutaten der Problemstellung**

4 Zutaten der Lösung

5 Die Zubereitung

6 Fazit und weitere Informationen.

3

Zutaten in der Architekturdokumentation

Problemstellung

- Mission Statement
- Kontextabgrenzung
- Architekturziele
- Entscheidende Randbedingungen
- Größte Risiken

Lösungsstrategie

- Lösungsstrategie
- Architekturprinzipien
- Informelles Überblicksbild („Big Picture“)

Lösungsdetails

- Architektur-entscheidungen
- Zerlegung
- Verhalten
- Verteilung
- Konzepte

Zutaten zur Problemstellung ...

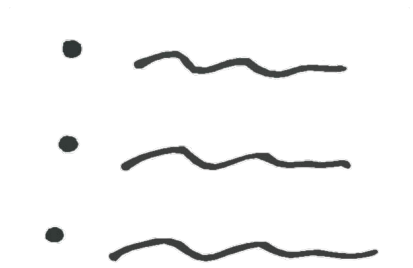
... grenzen die **Aufgabe** ab, beschreiben die **Ziele** des Systems und die zentralen **Einflussfaktoren** auf Eure Lösung.

Zutat	Format	Beschreibung
Mission Statement		Plakative Darstellung der Aufgabenstellung (Wozu ist das System da? Wem nützt es? ...)
Kontextabgrenzung		Visualisierung der Fremdsysteme und Benutzer, mit denen das System interagiert
Architekturziele (Top 3-5)		Die wichtigsten an das System gestellten Qualitätsanforderungen („-ilities“), inkl. Motivation
Entscheidende Randbedingungen		Die wichtigsten technischen bzw. organisatorischen Vorgaben, die beim Entwurf einzuhalten sind (oder waren)
Größte Risiken		Mögliche schädliche Ereignisse, die Einfluss auf die Softwarearchitektur haben (oder hatten).

Mission Statement

Plakative Darstellung der Aufgabenstellung.

Form: Stichpunkte oder wenige (!) Sätze.

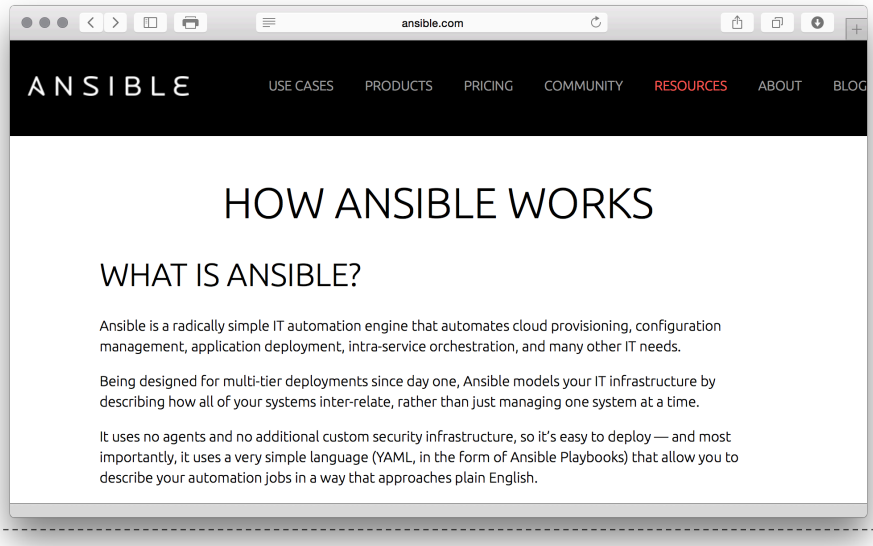


Leitfragen zur Erstellung

- Wozu ist das System (oder die Systemlandschaft, Komponente ...) da?
- Was ist das zentrale Verkaufs- / Nutzungsargument? (“Claim”, “Slogan”)
- Wem nützt es?
- Was sind die wesentlichen Features des Systems?
- Wie unterscheidet es sich von Produkten der Mitbewerber, oder der Vorgängerversion?

Beispiel: What is Ansible?

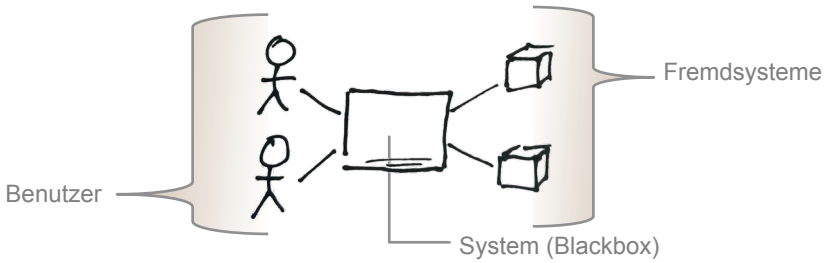
➔ http://www.ansible.com/how-ansible-works



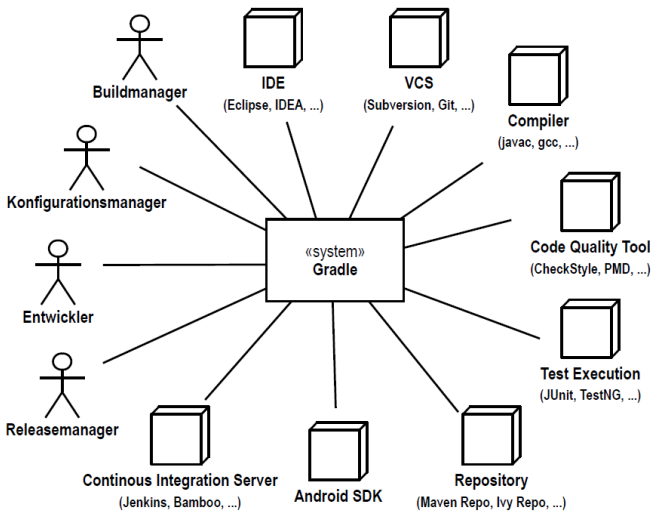
Kontextabgrenzung

Abgrenzung des Systems und Visualisierung der Fremdsysteme und Benutzer, mit denen es interagiert.

Form: Graphik, ergänzt um kurze Beschreibungen.



Beispiel



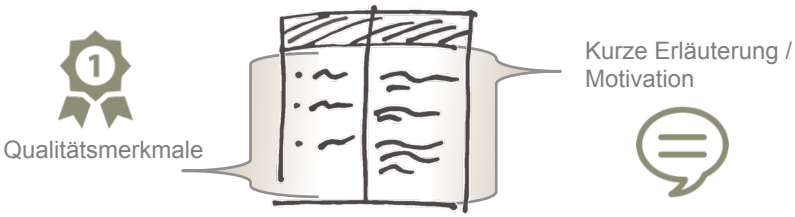
gradle

Quelle: Stefan Zörner, arc42-Startschnitt Gradle
http://update.hanser-fachbuch.de/2013/10/arc42-starschnitt-gradle-schnipsel-nr-2-systemkontext/

Architekturziele

Die wichtigsten (Top 3-5) an das System gestellten Qualitätsanforderungen („-ilities“), inkl. Motivation

Form: Stichpunkte oder (besser) Tabelle.



Qualitätsmerkmale ("–heiten und –keiten")

Beispiel

Ziel	Beschreibung
Verfügbarkeit	Die Lösung steht auch bei Lastspitzen uneingeschränkt zur Verfügung.
Benutzbarkeit	Auf allen wichtigen Geräten eine optimale User Experience.
Modifizierbarkeit	Es ist leicht, neue Funktionalität zu bauen und hinzuzufügen.
Attraktivität	Wir sind als Arbeitgeber für gute Entwickler attraktiv.
Betreibbarkeit	Wir haben gute Einblicke, was in unserer Umgebung läuft.

NETFLIX

Quelle: Stefan Toth, Stefan Zörner
"Gut das ist? Umgekehrte Architekturbewertung eines Internetgiganten"

Beispiel

Ziel	Beschreibung
Erweiterbarkeit	Gradle lässt sich leicht um neue Funktionalität erweitern. Es kann auf lange Sicht dem technologischen Fortschritt bei Tools und Entwicklungsmethodik folgen.
Effizienz	Teams und einzelne Entwickler erhalten durch kurze Buildzeiten schnelle Rückmeldungen; Gradle steigert so ihre Produktivität.
Interoperabilität	Gradle arbeitet mit bestehenden Werkzeugen wie Ant und Maven und deren Öko-Systemen nahtlos zusammen.
Installierbarkeit	Der Aufwand, der zum Installieren von Gradle notwendig ist, um einen Build auszuführen, ist sehr gering.

gradle

Quelle: Stefan Zörner, arc42-Startschnitt Gradle
http://update.hanser-fachbuch.de/2013/10/arc42-starschnitt-gradle-schnipsel-nr-3-qualitaetsziele/

Zutaten im Zusammenspiel

Problemstellung Lösungsstrategie Lösungsdetails

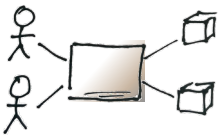
- **Mission Statement**
- **Architekturziele**
- **Randbedingungen**
- **Risiken**
- **Kontextabgrenzung**

Beeinflussen Architekturentscheidungen

Zutaten im Zusammenspiel

Problemstellung Lösungsstrategie Lösungsdetails

- **Mission Statement**
- Einflussfaktoren
  - **Architekturziele**
  - **Randbedingungen**
  - **Risiken**
- **Kontextabgrenzung**



Agenda

1 Motivation
2 Formen und Zutaten
3 Zutaten der Problemstellung
4 **Zutaten der Lösung**
5 Die Zubereitung
6 Fazit und weitere Informationen.

4

Zutaten in der Architekturdokumentation

Problemstellung

- Mission Statement
- Kontextabgrenzung
- Architekturziele
- Entscheidende Randbedingungen
- Größte Risiken

Lösungsstrategie

- Lösungsstrategie
- Architekturprinzipien
- Informelles Überblicksbild („Big Picture“)

Lösungsdetails

- Architektur-entscheidungen
- Zerlegung
- Verhalten
- Verteilung
- Konzepte

Zutaten zur Lösungsstrategie

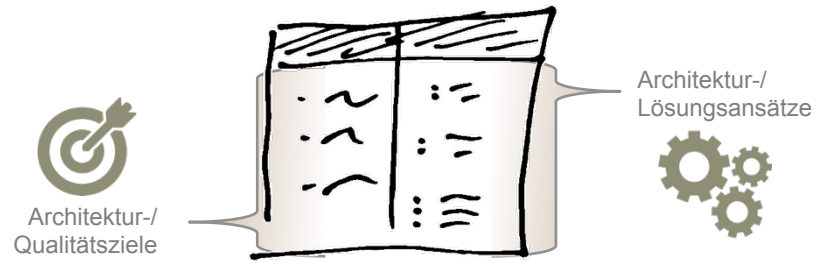
... schlagen die **Brücke** zwischen Problem und Lösung („The Big Picture“)

Zutat	Format	Beschreibung
Lösungsstrategie		Tabelle mit Architekturzielen und zugeordneten high-level Lösungsansätzen, mit Verweisen auf Überblicksbild und Lösungsdetails
Architektur-prinzipien		Grundsätze, an denen sich alle Entscheidungen orientieren. z.B. Präferenzen („Bevorzuge XY vor Z“)
Informelles Überblickbild		Visualisierung der Lösung mit Betonung der zentralen Architekturansätze (z.B. Stil, Muster, Struktur ...)

Lösungsstrategie (Tabelle)

Architekturziele und zugeordnete high-level Lösungsansätze, mit Verweisen auf Überblicksbild und Lösungsdetails

Form: Tabelle ([Ziele | Lösungsansätze]).



Typische Lösungsansätze

Beispiel

Architekturziel	passende Lösungsansätze in Gradle
Erweiterbarkeit	• Build-Files sind Groovy-Skripte, kleine Erweiterungen auch ohne aufwändige Programmierung möglich • Plugin-Konzept, Custom Plugins können in Groovy, Java, Scala … entwickelt werden • Exzellent dokumentierte Gradle API und erweiterbare DSL
Effizienz	• Unterstützung inkrementeller Builds • Dependency-Cache zur Reduktion der Download-Zeiten • Betrieb im Hintergrund für kürzere Start- und Ausführungszeiten möglich (Gradle Daemon)
Interoperabilität	• Direkte Verwendung von Ant-Tasks und Ant-Projekten in Gradle möglich • Konverter für Maven pom.xml nach Gradle Build-Skript verfügbar • Unterstützung von Maven-Repositories (Download von Abhängigkeiten, Veröffentlichen von Artefakten)

Quelle: Stefan Zörner, arc42-Startschnitt Gradle, http://update.hanser-fachbuch.de/2014/12/arc42-starschnitt-gradle-schnipsel-nr-10-loesungsstrategie/

Beispiel: Netflix (Ausschnitt)

Ziel	Passende Lösungsansätze
Verfügbarkeit	• Betrieb in Amazon Cloud • Circuit Breaker pattern
Benutzbarkeit	• Netflix Client API • Reative Extensions
Modifizierbarkeit	• Microservices • Komplette Automatisierung
Attraktivität	• Einsatz modernster Technologien • Viele Lösungen Open Source
…	…

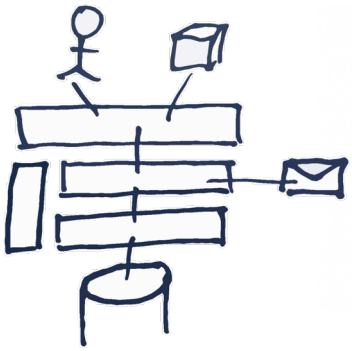
NETFLIX

Quelle: Stefan Toth, Stefan Zörner
"Gut das ist? Umgekehrte Architekturbewertung eines Internetgiganten"

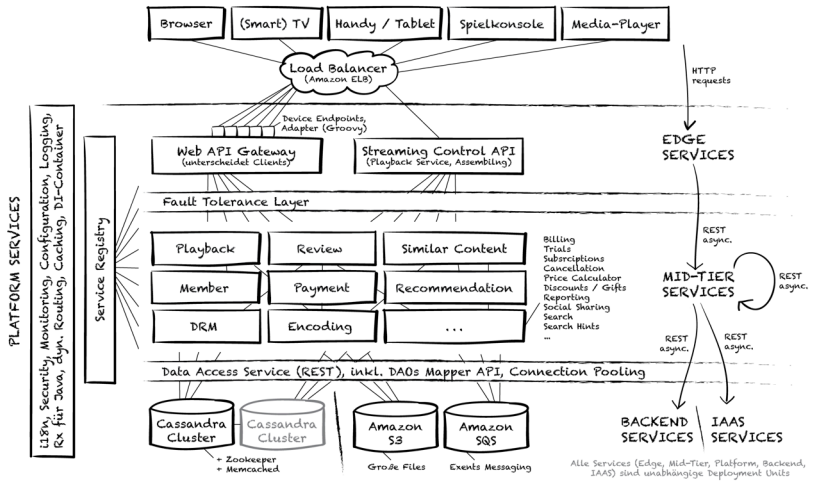
Informelles Überblicksbild

Visualisierung der Lösung mit Betonung der zentralen Architekturansätze (z.B. Stil, Muster, Struktur ...)

Form: Abbildung (eher kein UML ;-)



Beispiel

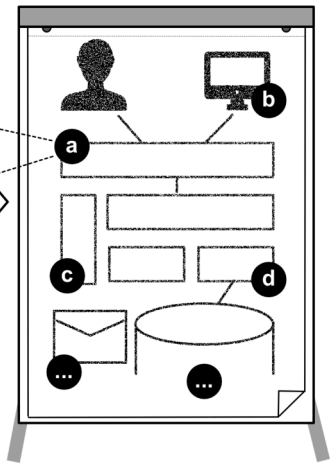


NETFLIX

Quelle: Stefan Toth, Stefan Zörner
"Gut das ist? Umgekehrte Architekturbewertung eines Internetgiganten"

Zusammenspiel: Tabelle und Bild

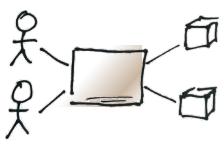
Qualitätsziele	Lösungsansätze
Ziel 1	* Ansatz (a) * Idee (b)
Ziel 2	* Konzept (c) * Ansatz (a) * Entscheidung (d)
...	* ...



Zutaten im Zusammenspiel

Problemstellung

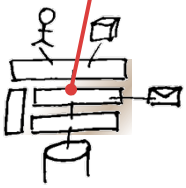
- Mission Statement
- Einflussfaktoren
  - Architekturziele
  - Randbedingungen
  - Risiken
- Kontextabgrenzung



Lösungsstrategie

- Lösungsstrategie

- Architekturprinzipien
- Überblicksbild



Zutaten in der Architekturdokumentation

Problemstellung

- Mission Statement
- Kontextabgrenzung
- Architekturziele
- Entscheidende Randbedingungen
- Größte Risiken

Lösungsstrategie

- Lösungsstrategie
- Architekturprinzipien
- Informelles Überblicksbild („Big Picture“)

Lösungsdetails

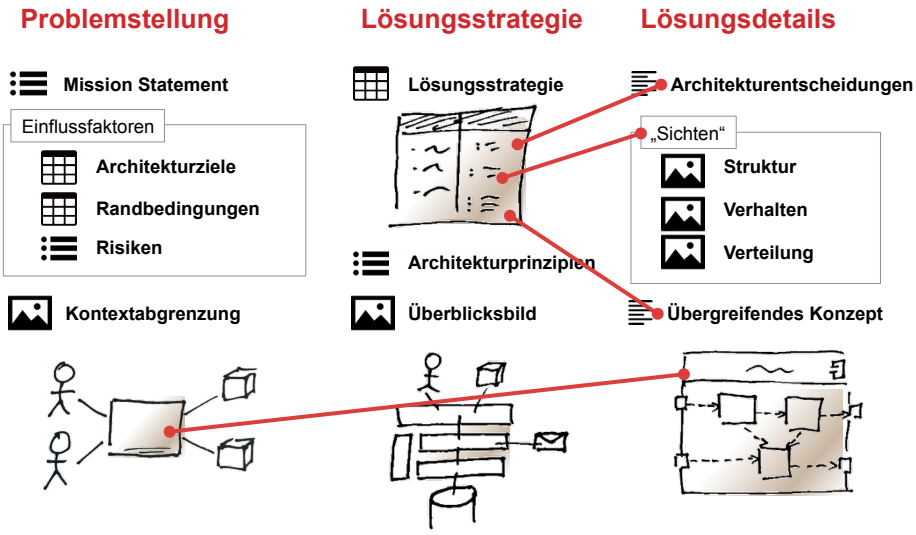
- Architektur-entscheidungen
- Zerlegung
- Verhalten
- Verteilung
- Konzepte

Zutaten zu Lösungsdetails ...

... Zeigen Lösungsansätze im Detail und machen die Architektur **nachvollziehbar** und **einhaltbar**.

Zutat	Format	Beschreibung
Architektur-entscheidung	≡	Nachvollziehbare Darstellung einer zentralen, risikoreichen Entscheidung, z.B. zum Technologie-Stack
Struktur	🖼	Technische und oder fachliche Zerlegung des Systems, z.B. anhand von Mustern
Verhalten	🖼	Zentrale Abläufe innerhalb des Systems („Walk Through“)
Verteilung	🖼	Visualisierung der Zielumgebung, der Inbetriebnahme und des Betriebs des Systems.
Technisches Konzept	≡	Darstellung einer systemübergreifenden Idee

Zutaten im Zusammenspiel



Agenda

1 Motivation
2 Formen und Zutaten
3 Zutaten der Problemstellung
4 Zutaten der Lösung
5 **Die Zubereitung**
6 Fazit und weitere Informationen.

5

Schritt für Schritt zum Überblick ...

1. Zielgruppen und Ziele identifizieren
2. Zutaten auswählen und anfertigen
3. Zutaten in Form bringen
4. Architekturüberblick auf Gebrauchstauglichkeit überprüfen

Schritt für Schritt zum Überblick

1 Zielgruppen und Ziele identifizieren

Zum Beispiel Neue im Team, Entscheider, teamfremde Kollegen ...

2 Zutaten auswählen und anfertigen

Passend zu den Zielgruppen und deren Zielen

In der Kreuztabelle festhalten, welche Zutat primär für welche Zielgruppen ist. Schreiben Sie aus Sicht der Leser.

3 Zutaten in Form bringen

Mit Hilfe der Zielgruppen geeignete Formen wählen und Zutaten darin zu Überblicken kombinieren

Statt eines bloßen X können Sie in den Zellen der Kreuz- tabelle auch Dringlichkeit und/ oder nötige Tiefe (detailliert, Überblick ...) eintragen.

4 Architekturüberblick auf Gebrauchstauglichkeit überprüfen

Review durch Zielgruppe, Feedback einfließen lassen ...

Schritte wiederholen, bei Bedarf verfeinern, neue Zielgruppen, Formen, Zutaten ergänzen.

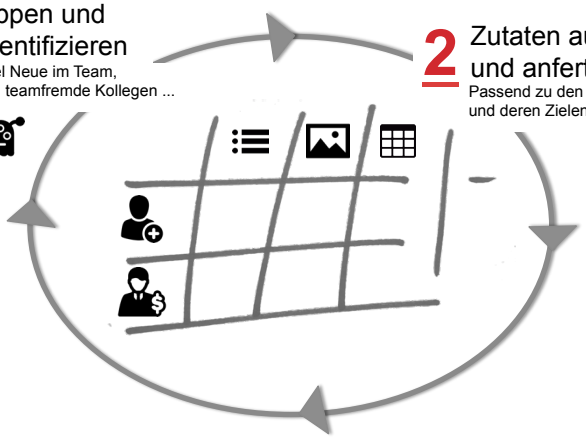
Schritt für Schritt zum Überblick

1 Zielgruppen und Ziele identifizieren

Zum Beispiel Neue im Team, Entscheider, teamfremde Kollegen ...

2 Zutaten auswählen und anfertigen

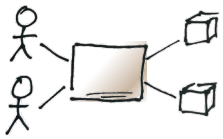
Passend zu den Zielgruppen und deren Zielen



Was gehört mindestens hinein?

Problemstellung

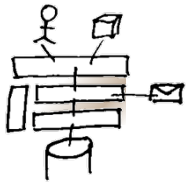
- **Mission Statement**
- Einflussfaktoren
  - **Architekturziele**
  - **Randbedingungen**
  - **Risiken**
- **Kontextabgrenzung**



Lösungsstrategie

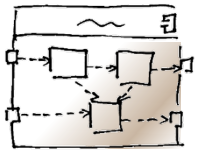
- **Lösungsstrategie**

- **Architekturprinzipien**
- **Überblicksbild**



Lösungsdetails

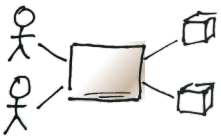
- **Architekturentscheidungen**
- „Sichten“
  - **Struktur**
  - **Verhalten**
  - **Verteilung**
- **Übergreifendes Konzept**



Was gehört mindestens hinein?

Problemstellung

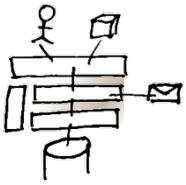
- **Mission Statement**
- Einflussfaktoren
  - **Architekturziele**
  - Randbedingungen
  - Risiken
- **Kontextabgrenzung**



Lösungsstrategie

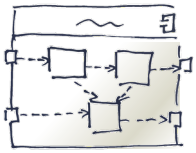
- **Lösungsstrategie**

- Architekturprinzipien
- **Überblicksbild**



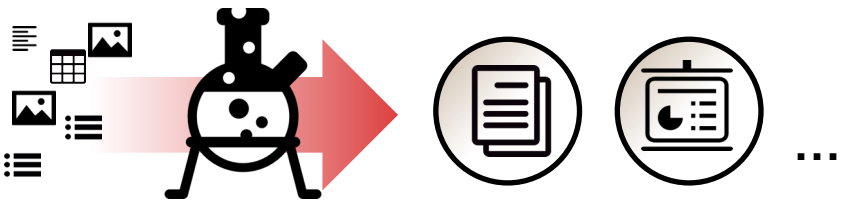
Lösungsdetails

- Architekturentscheidungen
- „Sichten“
  - Struktur
  - Verhalten
  - Verteilung
- Übergreifendes Konzept



Schritt für Schritt zum Überblick ...

1. Zielgruppen und Ziele identifizieren
2. Zutaten auswählen und anfertigen
3. **Zutaten in Form bringen**
4. Architekturüberblick auf Gebrauchstauglichkeit überprüfen



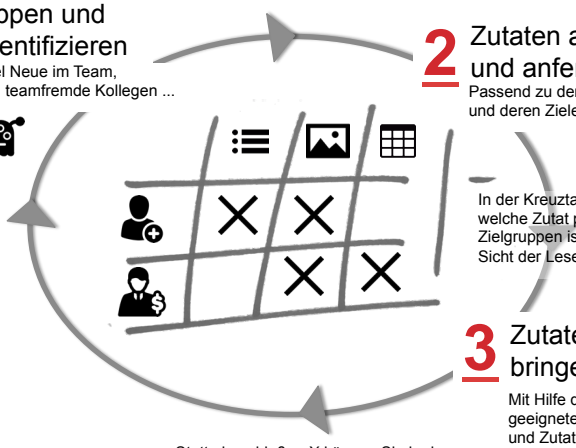
Schritt für Schritt zum Überblick

1 Zielgruppen und Ziele identifizieren

Zum Beispiel Neue im Team, Entscheider, teamfremde Kollegen ...

2 Zutaten auswählen und anfertigen

Passend zu den Zielgruppen und deren Zielen



In der Kreuztabelle festhalten, welche Zutat primär für welche Zielgruppen ist. Schreiben Sie aus Sicht der Leser.

3 Zutaten in Form bringen

Mit Hilfe der Zielgruppen geeignete Formen wählen und Zutaten darin zu Überblicken kombinieren

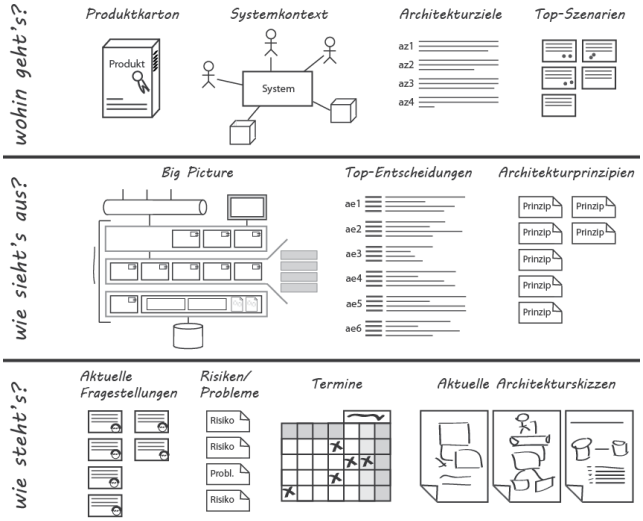
Statt eines bloßen X können Sie in den Zellen der Kreuztabelle auch Dringlichkeit und/ oder nötige Tiefe (detailliert, Überblick ...) eintragen.

Struktur für Folienvortrag

Abschnitt	Mögliche Inhalte
I. Aufgabenstellung	**Mission Statement** **Architekturziele** **Kontextabgrenzung** Herausforderungen, Schmerzpunkte Zentrale Randbedingungen
II. „The Big Picture“	**Lösungsstrategie** (Tabelle) Informelles Überblicksbild Architekturprinzipien
III. Die Lösung im Detail	**Architekturentscheidungen** Diagramme (Struktur, Verteilung, ...) Übergreifende Konzepte Demo, Walkthrough
IV. Fazit und Ausblick	**Offene Punkte** **Nächste Schritte** Diskussion Weitere Informationen Was sind Ihre Fragen?

(Fett gedruckte Inhalte sind besonders wichtig.)

Architekturwand

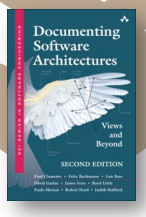


Quelle: S. Toth, Vorgehensmuster für Softwarearchitektur

Schritt für Schritt zum Überblick ...

1. Zielgruppen und Ziele identifizieren
2. Zutaten auswählen und anfertigen
3. Zutaten in Form bringen
4. **Architekturüberblick auf Gebrauchstauglichkeit überprüfen**

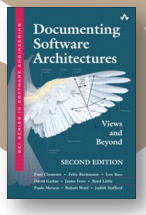
7 Regeln für gute Dokumentation



„Documenting Software Architectures: Views and Beyond“
Clements, et.al, 2. Auflage 2010

1. Schreibe aus Sicht des Lesers
2. Vermeide unnötige Wiederholungen
3. Vermeide Mehrdeutigkeiten
   3. a) Erkläre Deine Notation
4. Verwende eine Standardstrukturierung
5. Halte Begründungen für Entscheidungen fest
6. Halte Dokumentation aktuell, aber auch nicht zu aktuell
7. Überprüfe Dokumentation auf ihre Gebrauchstauglichkeit

7 Regeln für gute Dokumentation



„Documenting Software Architectures: Views and Beyond“
Clements, et.al, 2. Auflage 2010

1. Schreibe aus Sicht des Lesers
2. Vermeide unnötige Wiederholungen
3. Vermeide Mehrdeutigkeiten
   3. a) Erkläre Deine Notation
4. Verwende eine Standardstrukturierung
5. Halte Begründungen für Entscheidungen fest
6. Halte Dokumentation aktuell, aber auch nicht zu aktuell
7. **Überprüfe Dokumentation auf ihre Gebrauchstauglichkeit**

Schritt für Schritt zum Überblick

1 Zielgruppen und Ziele identifizieren

Zum Beispiel Neue im Team, Entscheider, teamfremde Kollegen ...

2 Zutaten auswählen und anfertigen

Passend zu den Zielgruppen und deren Zielen

In der Kreuztabelle festhalten, welche Zutat primär für welche Zielgruppen ist. Schreiben Sie aus Sicht der Leser.

3 Zutaten in Form bringen

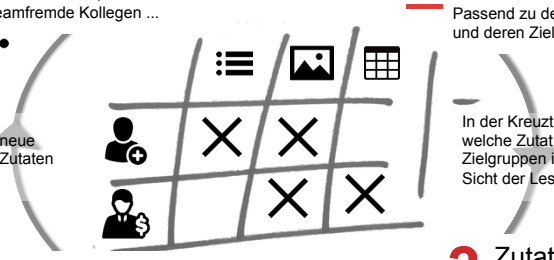
Mit Hilfe der Zielgruppen geeignete Formen wählen und Zutaten darin zu Überblicken kombinieren

Statt eines bloßen X können Sie in den Zellen der Kreuztabelle auch Dringlichkeit und/ oder nötige Tiefe (detailliert, Überblick ...) eintragen.

4 Architekturüberblick auf Gebrauchstauglichkeit überprüfen

Review durch Zielgruppe, Feedback einfließen lassen ...

Schritte wiederholen, bei Bedarf verfeinern, neue Zielgruppen, Formen, Zutaten ergänzen.



Agenda

6

1 Motivation

2 Formen und Zutaten

3 Zutaten der Problemstellung

4 Zutaten der Lösung

5 Die Zubereitung

6 **Fazit und weitere Informationen.**

tl;dr -- "Too long; didn't read"

> *Ein **Architekturüberblick** macht die zentralen **Lösungsansätze** Eurer Softwarearchitektur für Außenstehende **nachvollziehbar**.*

> *Wenn Ihr kleinteilig arbeitet (Stichwort **„Zutaten“**), habt Ihr schnell etwas Vorzeigbares. Zentral: Die **Lösungsstrategie** als Brückenschlag.*

> *Die **Zutaten** lassen sich dann **zielgruppengerecht** in unterschiedlichen **Formen** (Flyer, Dokumente, ...) rekombinieren.*

Buch: Softwarearchitekturen dokumentieren und kommunizieren

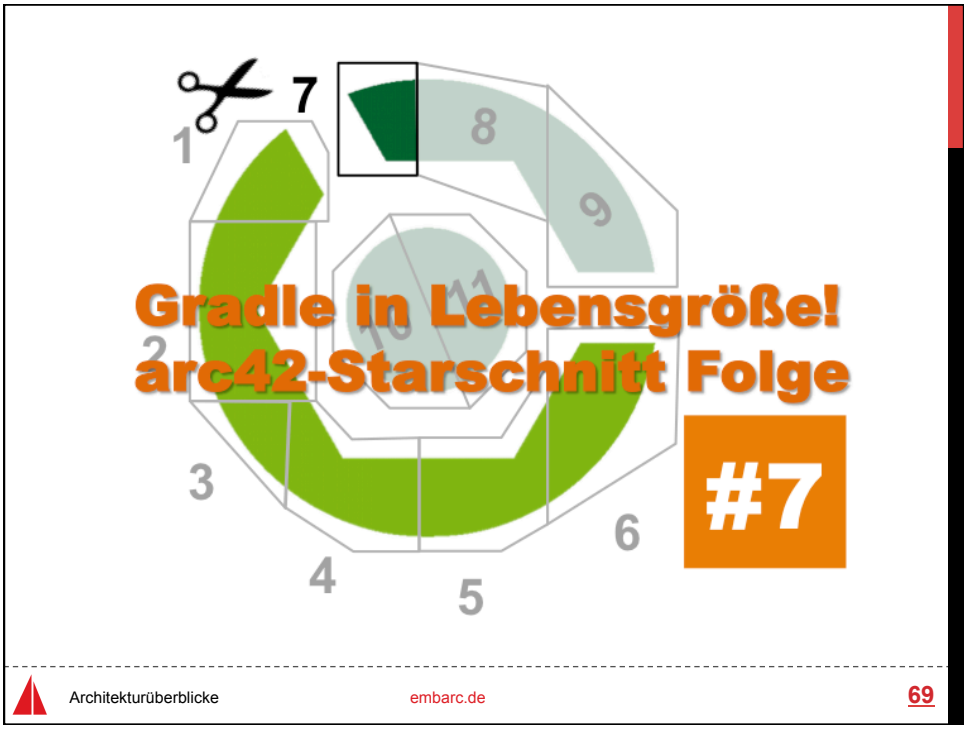
Entwürfe, Entscheidungen und Lösungen nachvollziehbar und wirkungsvoll festhalten

Autor: Stefan Zörner

Umfang: ca. 280 Seiten

Verlag: Hanser Verlag, 2. Aufl., Mai 2015
Sprache: Deutsch
ISBN: 978-3-446-42924-6

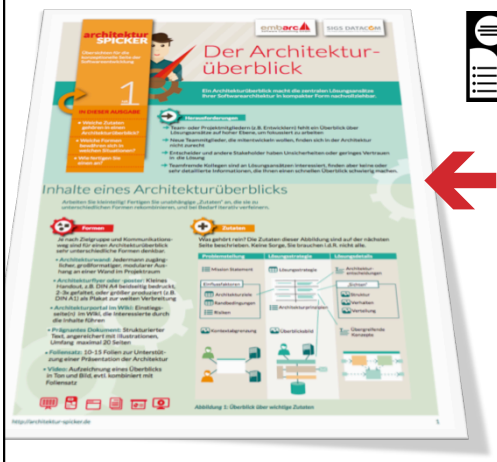
www.swadok.de



Blog-Serie bei Hanser Update

➔ update.hanser-fachbuch.de/tag/arc42-starschnitt/

Spicken erlaubt!



Unsere Architektur-Spicker beleuchten die konzeptionelle Seite der Softwareentwicklung.

Spicker #1:
„Der Architekturüberblick“

- Welche Zutaten gehören in einen Architekturüberblick?
- Welche Formen bewähren sich in welchen Situationen?
- Wie fertigen Sie einen Architekturüberblick an?

PDF, 4 Seiten
Kostenloser Download.

➔ http://architektur-spicker.de

Vielen Dank.

Ich freue mich auf Eure Fragen!

stefan.zoerner@embarc.de
@StefanZoerner
xing.to/szr

embarc Software Consulting GmbH

DOWNLOAD FOLIEN:
http://www.embarc.de/blog/