

Stefan Zörner

Die LASR-Methode für Software-Reviews ... in 20 Minuten erklärt.

Vortrag bei den InfoDays: Software-Architektur
powered by Software Architecture Alliance
Remote, 07. Oktober 2026



embarc.de

LASR-Methode für Software-Reviews

1

Abstract

Die LASR-Methode für Software-Reviews ... in 20 Minuten erklärt.

Architekturbewertungen sichern Lösungsideen ab und decken Risiken in Entwurf und Umsetzung auf. Klassische Analyseansätze aus diesem Umfeld wie ATAM sind fundiert, kommen aber gerade in beweglichen Softwarevorhaben etwas akademisch und schwergewichtig, mitunter fast zeremoniell daher.

Mit dem Lightweight Approach for Software Reviews (LASR) gewinnen Entwicklungsteams in kurzer Zeit erste, kommunizierbare Erkenntnisse. Hier lernt ihr den Ablauf und das frei verfügbare Unterstützungsmaterial von LASR kennen und seht Beispiel-Ergebnisse aus dessen Anwendung. Im Anschluss könnt ihr direkt loslegen, und ein pointiertes Review zielgerichtet an einem halben Tag durchspielen.



embarc.de

Softwarearchitektur in Eigenregie bewerten

2

Stefan Zörner

- Softwarearchitekt bei embarc in Hamburg
- Vorher Bayer AG, Mummert + Partner, IBM, oose, ...

Schwerpunkte

- Methodische Softwarearchitektur (Entwurf, Bewertung, Dokumentation)
- Architektur-Reviews



01.

Warum Reviews?

Wozu sind Software-Reviews da und welche Ansätze gibt es grundsätzlich dazu?





Typische Fragestellungen für Reviews

Eine **Neuentwicklung** steht an und erste Lösungsansätze stehen im Raum.



Leitfrage im Review

Seid ihr und euer Team auf dem **richtigen Weg**?

Größere **Umbaumaßnahmen** in eurer Software stehen an.



Leitfrage im Review

Wie wählt ihr passende Lösungsansätze **nachvollziehbar** aus?



Architektur-Reviews **sichern Lösungsideen ab** und **decken Risiken** in Entwurf und Umsetzung **auf**.



Grundsätzliche Ansätze



Quantitative Analyse

Setzt auf Messungen und Metriken.
In der Regel **Tool-basiert**.



Qualitative Analyse

Setzt auf Diskussion, Austausch und Durchsprachen.
Oft **Workshop-basiert**.



Bewertungsmethoden (Auswahl)

SAAM Software Architecture Analysis Method	ARID Architecture Review for Intermediate Designs
ATAM Architecture Tradeoff Analysis Method	DCAR Decision Centric Architecture Review
CBAM Cost-Benefit Analysis Method	DASE Decision and Scenario based architecture evaluation
TARA Tiny Architecture Review Approach	Pre-Mortem Risk-Brainstorming and Mitigation
PBAR Pattern Based Architecture Review	LASR Lightweight Approach for Software Reviews



Bewertungsmethoden (Auswahl)

SAAM Software Architecture Analysis Method	ARID Architecture Review for Intermediate Designs
ATAM Architecture Tradeoff Analysis Method	DCAR Decision Centric Architecture Review
CBAM Cost-Benefit Analysis Method	DASE Decision and Scenario based architecture evaluation
TARA Tiny Architecture Review Approach	Pre-Mortem Risk-Brainstorming and Mitigation
PBAR Pattern Based Architecture Review	LASR Lightweight Approach for Software Reviews



embarc.de

LASR-Methode für Software-Reviews



Herausforderungen ...

Die **Anwendung** fundierter Bewertungsmethoden wie ATAM ist **mitunter schwierig**.

- Der Einsatz erfordert häufig **viele Beteiligte**.
- Es sind oftmals **Vorarbeiten nötig**, beispielsweise die Aufbereitung der Geschäftsziele und das Anfertigen eines Architekturüberblicks.
- Sie liefern **nur Rohergebnisse**, die für eine effiziente Kommunikation aufwendig nachzubearbeiten sind.
- Durchführung und Moderation verlangen einiges ab – die Methoden **unterstützen** dabei **wenig**.



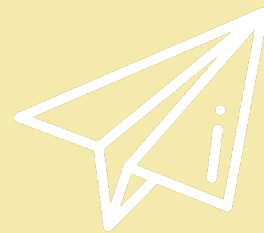
02.

LASR

Was ist dieses LASR und was macht es anders oder besonders?



Was ist leichtgewichtig?

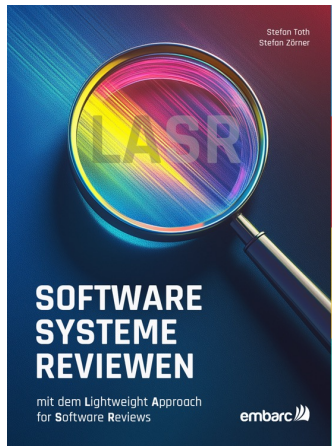


Merkmale eines leichtgewichtigen Reviews

- Die **Anzahl** der Beteiligten / **Stakeholder** ist **gering**.
- **Aufwand** und Dauer sind **überschaubar**.
- Erste **Ergebnisse** liegen vergleichsweise **schnell** vor.



Erfahrungswissen



Software-Systeme reviewen mit dem Lightweight Approach for Software Reviews - LASR



Autoren: Stefan Toth, Stefan Zörner
Verlag: Leanpub, September 2023
Sprache: Deutsch, EPUB, PDF

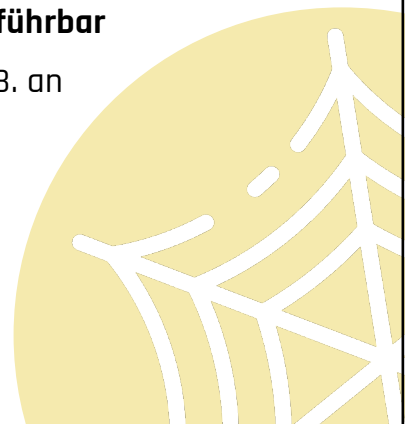
→ leanpub.com/software-systeme-reviewen/



Highlights von LASR

LASR (**L**ightweight **A**pproach for **S**oftware-**R**eviews) ist ein strukturiertes Vorgehen für leichtgewichtige Software-Reviews.

- Mit dem **eigenen Team** und potentiell **alleine durchführbar**
- Liefert **schnell** ein **erstes, vorzeigbares Ergebnis**, z.B. an einem Nachmittag
- **Spinnennetzgraphik** zur Erkenntnisverdichtung und -kommunikation
- **Unterstützungsmaterial** für Bewertungsmaßstab, Risikofindung und Ergebniserarbeitung



03.

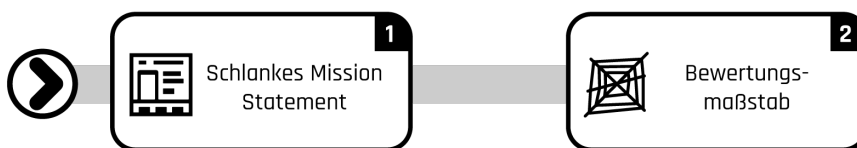
Review-Ablauf

In welchen Schritten läuft ein Review in LASR ab? Wie sieht das konkret aus?



Schritte im LASR-Review

Verstehe was Dich speziell macht



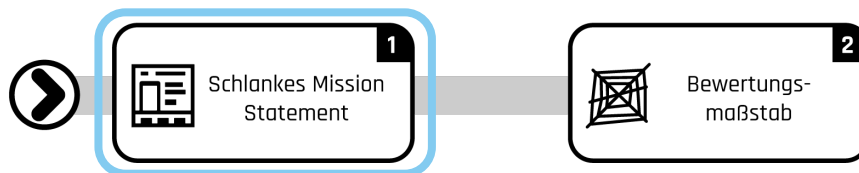
Durchleuchte die Architektur





Schritte im LASR-Review

Verstehe was Dich speziell macht



Durchleuchte die Architektur



(1) Schlankes Mission Statement



Fokussiert und **grenzt** das zu betrachtende System **ab** durch Finden sogenannter "Claims" (Ansprüche) der Software.

Methodischer Ansatz

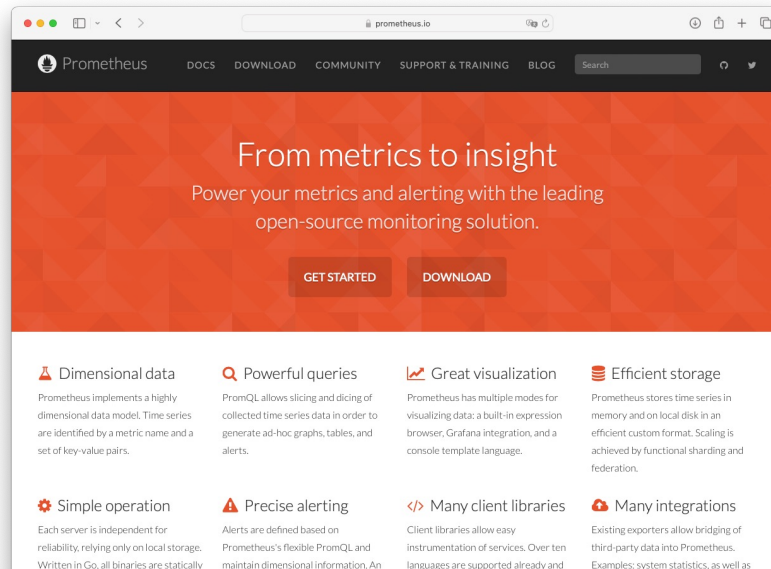
Sammeln der Claim(s) mit **Landing Page-Metapher**

Ergebnis des Schrittes

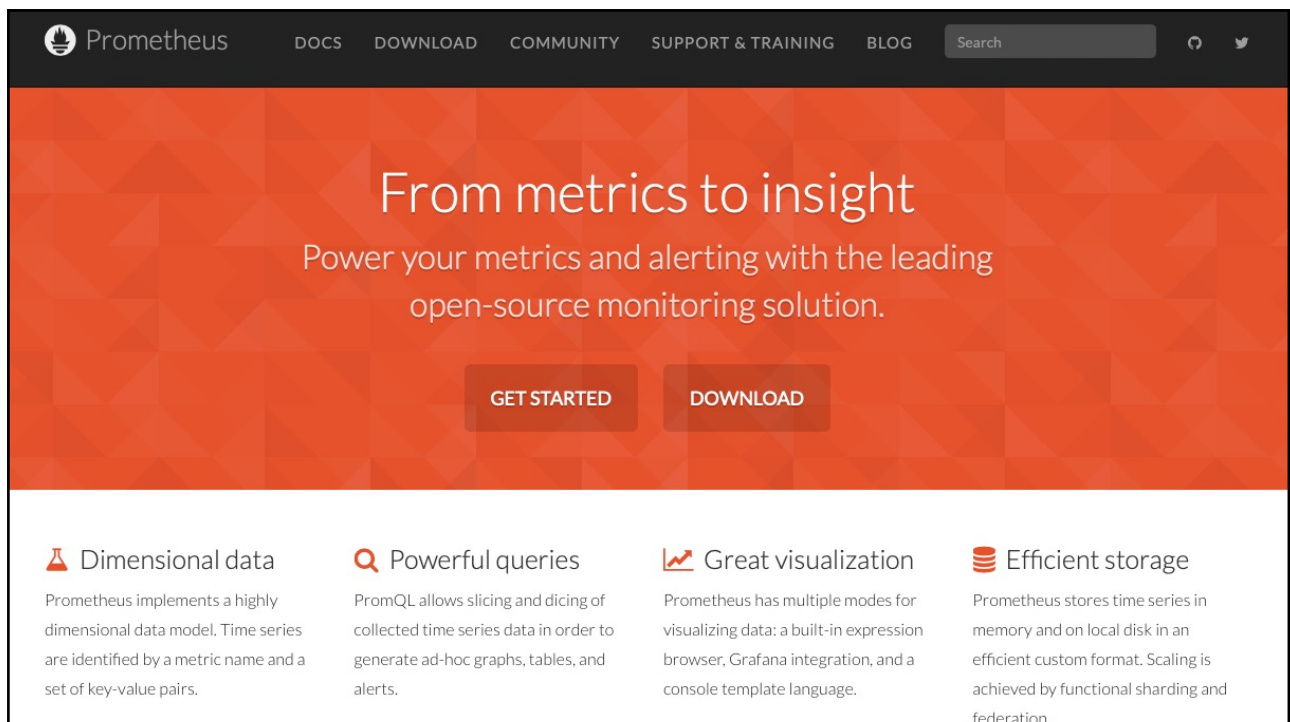
Prägnante Produktidee (Stichpunkte)



Beispiel: Landing Page von Prometheus



→ prometheus.io



Erarbeiten im Workshop

LASR Schritt 1: Schlankes Mission Statement

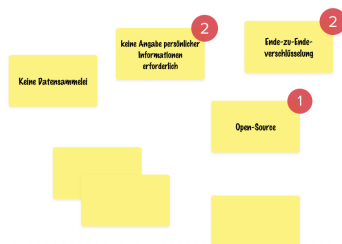
Was würde prominent auf der Landing Page der zu betrachtenden Softwarelösung stehen?



1 Brainstorming

Leitfragen:

Was sind zentrale Features oder Eigenschaften unserer Software?
Was macht unsere Software besonders? Was besonders gut?
Welche Ansprüche haben wichtige Beteiligte an die Softwarelösung? ("Claims")



2 Clustern und Ausdünnen

Voting. Welche der gefundenen Punkte sollten Eurer Meinung nach auf jeden Fall auftauchen? Ziel: ca. 7 +/- 2.

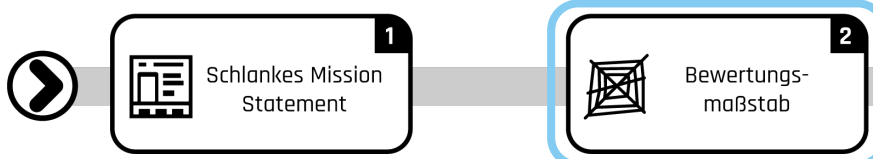


← Beispiel im digitalen Whiteboard (Mural)

Quelle: S. Toth, S. Zörner:
„Software-Systeme
reviewen“, Leanpub 2023

Schritte im Kern-Review

Verstehe was Dich speziell macht



Durchleuchte die Architektur





(2) Bewertungsmaßstab



Identifiziert und priorisiert grob die entscheidenden **Qualitätsmerkmale**.

Methodischer Ansatz

Top-5-Challenger zur Zielauswahl

Ergebnisse des Schrittes

3-5 Qualitätsziele inklusive grober Ziel-Einschätzung

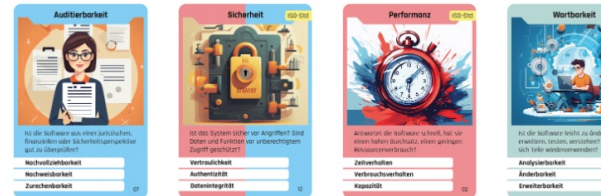




Top-5-Challenger (Unterstützungsmaterial)

Material

- 14 Karten mit Qualitätsmerkmalen



Vorbereitung

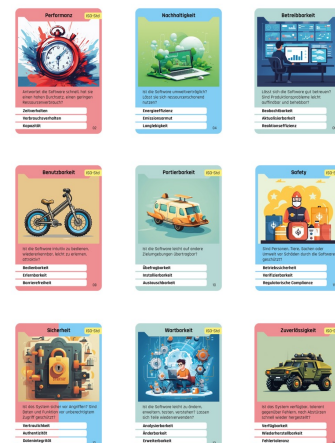
- Mischt die Karten in einem verdeckten Stapel.
- Deckt 5 Karten zufällig auf und legt sie nebeneinander. (das ist Eure Start Top-5)
- Legt die übrigen 9 in einer 3 x 3 Matrix für alle gut sichtbar offen aus.



Top-5-Challenger, Vorbereitung (Beispiel)



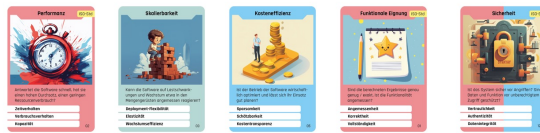
Start Top-5



Kandidaten



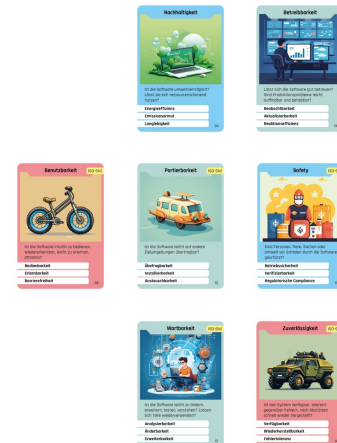
Nach 2 Runden, Beispiel (Beispiel)



Aktuelle Top-5



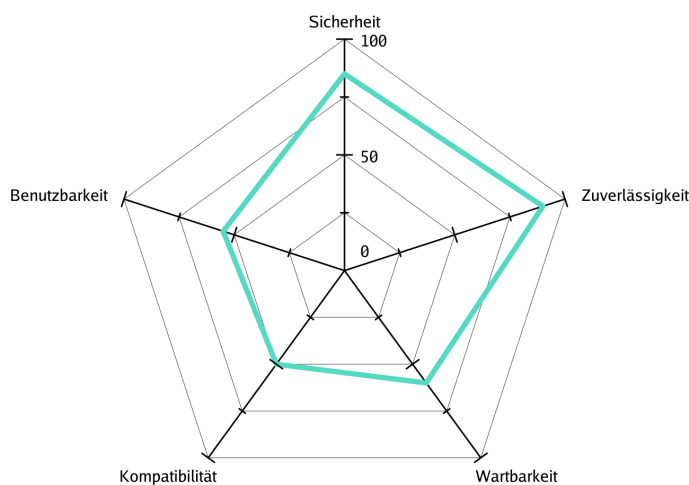
Ablagestapel



Kandidaten



Bewertungsmaßstab



LASR-Ergebnisdiagramm

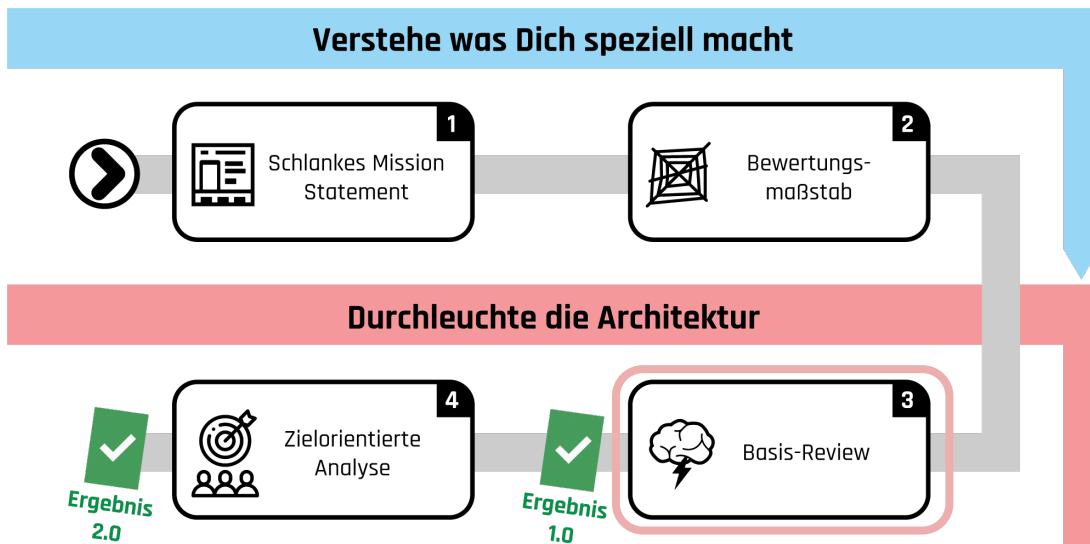
Die Top-3-5 **Qualitätsziele** bilden die **Achsen** des Diagramms.

Die **Zielwerte** spannen darauf eine **grüne Linie** auf.

← Beispiel



Schritte im Kern-Review



(3) Basis-Review



Produziert ein erstes Ergebnis in Form konkreter **Risiken**, die der Zielerreichung im Weg stehen.

Methodischer Ansatz

Brainstorming, angelehnt an **Pre-Mortem**

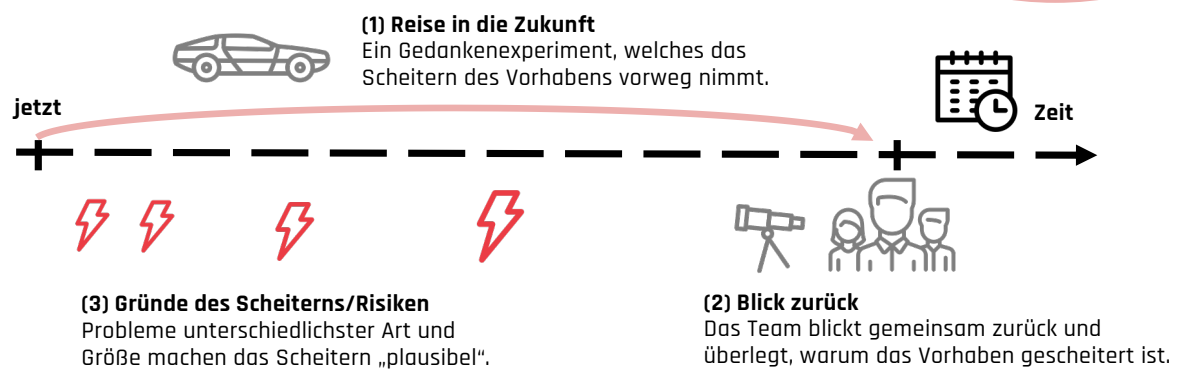
Ergebnisse des Schrittes

"Abweichung" vom Ziel, Ist-Einschätzung

Grundidee: Pre Mortem



Was ist ein Pre-Mortem?





Die 8 Risiko-Kategorien von LASR



Softwarelösung 1

- Unpassende Technologien
- Komplexe Lösungen
- Unausgereifte Fremdlösungen
- Behindernde Frameworks / Libraries
- Strukturprobleme
- ...



Kompetenz und Erfahrung 2

- Fehlendes oder isoliertes Wissen
- Zu kleine Lernfenster
- Fehlendes Prozessverständnis
- Tool-Probleme
- Schwere Schätzbarkeit
- ...



Zielsetzungen und Erwartungen 3

- Unrealistische Ziele
- Überzogene Erwartungen
- Knappe Deadlines
- Instabile Anforderungen
- Fehlender Kundenkontakt
- ...



Fremdsysteme und Plattformen 4

- Instabile Plattformen
- Fehleranfällige Fremdsysteme
- Unpassende Buy/Make Wahl
- Lizenzschwierigkeiten
- Vendor Lock-in
- Integrationsprobleme
- ...



Altsysteme und Altlasten 5

- Legacy-Behinderungen
- Innovationsstau
- Zerbrechliche Lösungen
- Fehlende Tests
- Wenig Dokumentation
- Fehlendes Verständnis
- ...



Organisation und Prozesse 6

- Geschwollene Prozesse
- Behindernde Rollenmodelle
- Organisationsgrenzen
- Politik
- Einengende Standards
- Isolierte Entscheidungskompetenzen
- ...



Betrieb und Deployment 7

- Blockierende Prozesse
- Geringe Automatisierung
- Wenig Feedback
- Fehlender Betriebes- / Ausfallkonzepte
- Tool-Probleme
- ...



Weiche Faktoren 8

- Uneingetriggten
- Konflikte
- Fehlende Disziplin
- Kommunikationsbarrieren
- Rollenethemiken
- Unpassende Kultur
- ...



Typische Risikothemen als Ideengeber



Zu hohe Komplexität der Lösung

Hat die Domäne eine sehr hohe Komplexität? Gibt es überlegte, schnelle Lösungen oder fehlende Abstraktionen? Komplexität gefährdet den Überblick bzw. Wartbarkeit, Korrektheit, Sicherheit, ...

Softwarelösung 1.1



Unpassende Lösungs-Strukturierung

Folgt die Softwarestruktur der Domäne? Sind andere technische oder organisatorische Einflüsse (Conway...) gut aufgegriffen? Falls nicht könnte Wartbarkeit, Zuverlässigkeit etc. leiden.

Softwarelösung 1.2



Inadäquates Daten-Handling

Ist die Ablage, der Transport und das Mapping von Daten konzeptionell und technisch passend gelöst? Sind Daten ausreichend schnell und rechtssicher, im richtigen Format an den richtigen Stellen?

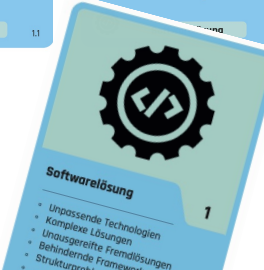
Softwarelösung 1.3



Problematische Konzepte & Technologien

Passen die eingesetzten Muster und Konzepte zu den Zielen? Sind sie konsistent angewendet? Sind die verwendeten Technologien und Frameworks passend und etabliert?

Softwarelösung 1.4



Softwarelösung

1

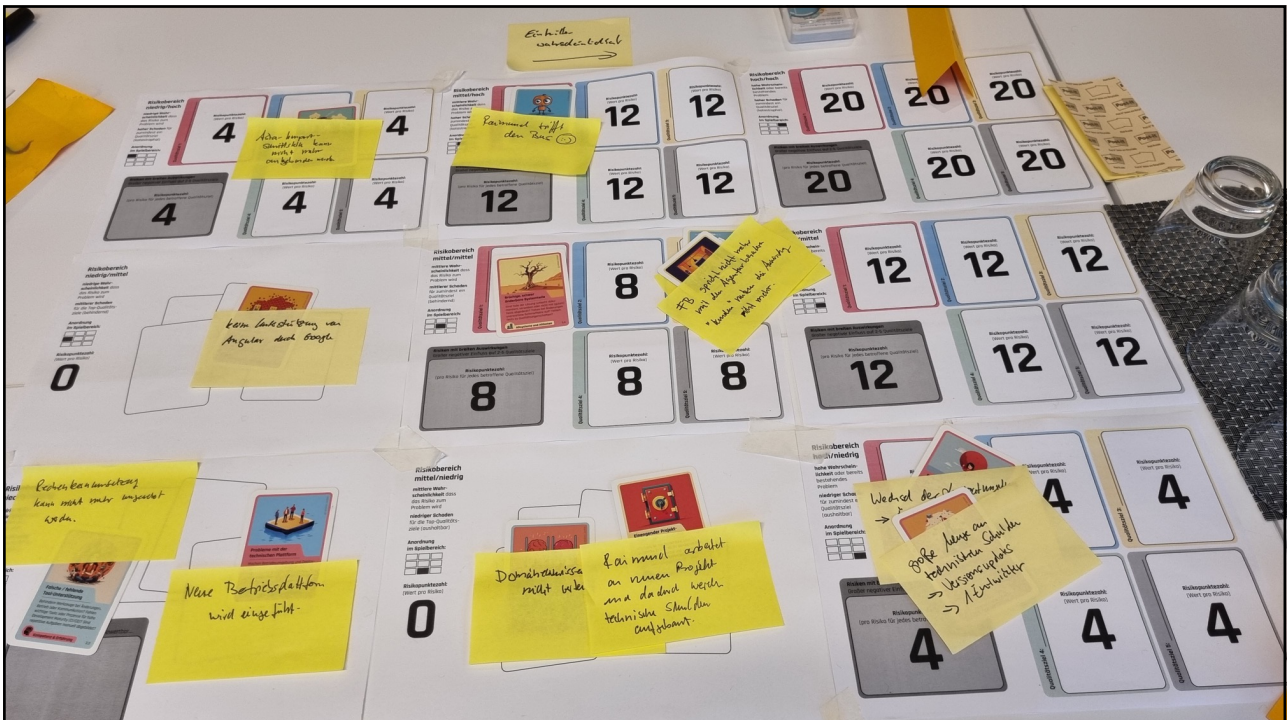
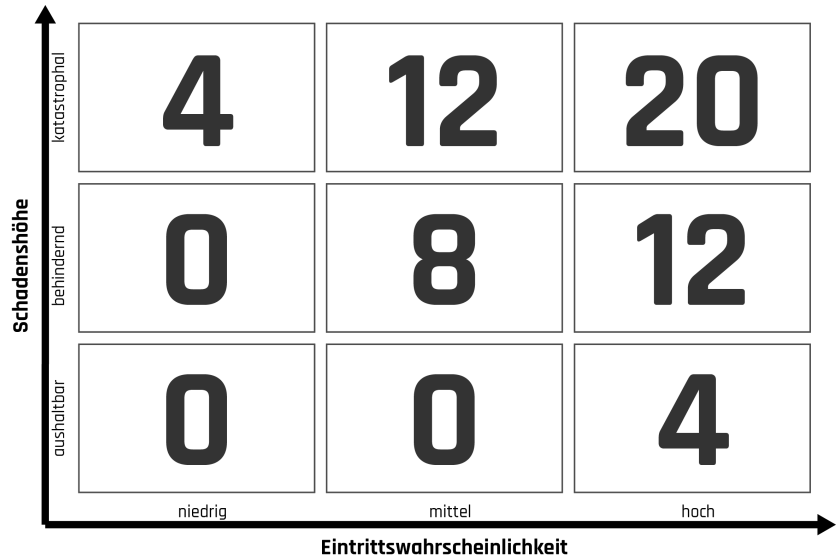
- Unpassende Technologien
- Komplexe Lösungen
- Unausgereifte Fremdlösungen
- Behindernde Frameworks
- Strukturprobleme

Brainstorming-Unterstützung: Das LASR-Kartenset hält insgesamt 32 konkrete Risikokarten als **Ideengeber** parat, 4 in jeder der 8 Kategorien.





Risiken einordnen



04.

Ein erstes Ergebnis

Wie sieht das Review-Ergebnis aus? Was machen wir damit?



Von Risiken zu „Lücken“



28% Abweichung
vom **Ziellevel**
auf der **Qualitätszielachse**

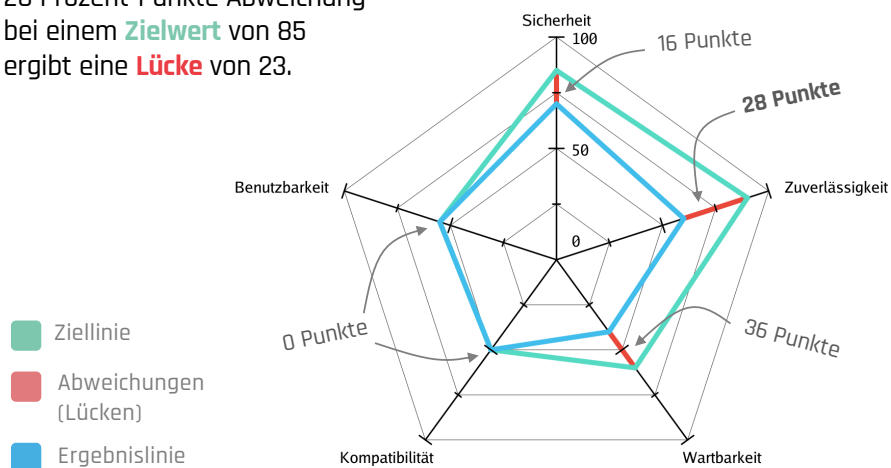




Abweichungen einzeichnen

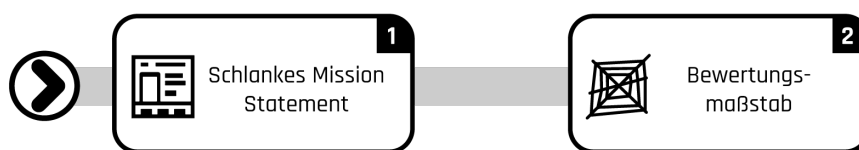
Beispiel

28 Prozent-Punkte Abweichung bei einem **Zielwert** von 85 ergibt eine **Lücke** von 23.



Basis-Review, 1. Ergebnis nach Schritt 3

Verstehe was Dich speziell macht

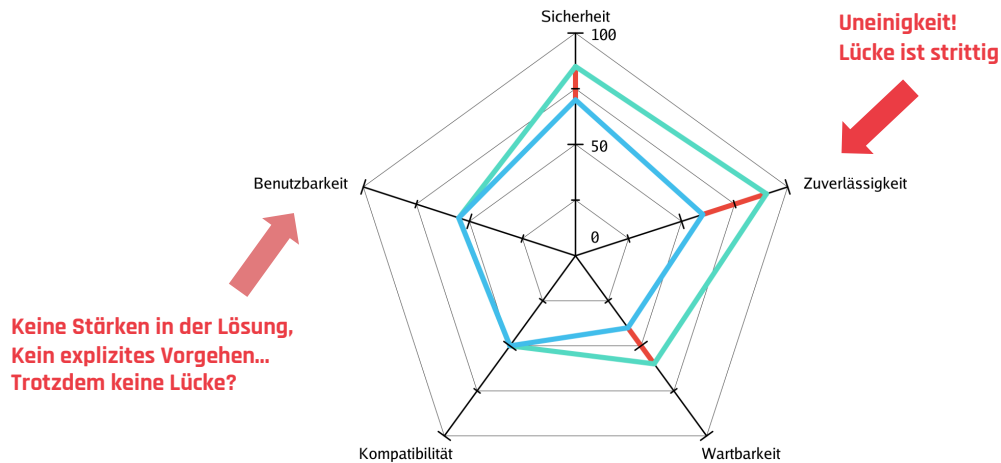


Durchleuchte die Architektur

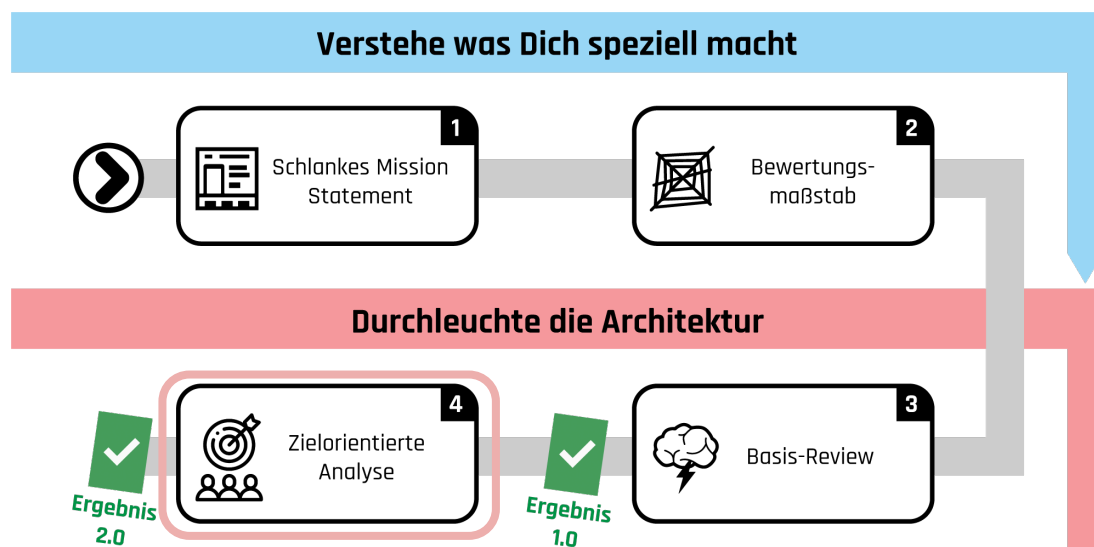




Das erste Ergebnis diskutieren



Schritte im Kern-Review



(4) Zielorientierte Analyse



Untersucht **gezielt** Stärken und Schwächen im System und **schärft** so das Ergebnis.

Methodischer Ansatz

Qualitative Durchsprache "strittiger" Ziel-Achsen

Ergebnisse des Schrittes

verbesserte Ist-Einschätzung, Qualitätsaussagen



LASR an einem Vormittag

Review-Workshop System YX -- Agenda

Die Zeiten dienen Euch zur groben Orientierung.
Wir machen aber pünktlich Schluss.



09:00 - 10:00

Begrüßung,
Einstieg LASR

ca. 10 Minuten

1
Abgrenzung,
Mission Statement
("Claims")

ca. 15-20 Minuten

2
Bewertungsmaßstab
festlegen

ca. 20-30 Minuten

kurze Pause
(+ Basis-Review
vorbereiten)

10:15 - 11:30

3
Basis-Review
("Pre-Mortem")

ca. 60 Minuten

3
Zielerreichung
bestimmen

ca. 15 Minuten

kurze Pause

11:45 - 12:15

3
Ergebnisse
diskutieren

ca. 10 Minuten

3
Ziel-orientierte
Analyse planen,
TODOs

ca. 10-20 Minuten

Ende des Workshops

(c) embarc GmbH

← Beispiel-Agenda

Quelle: S. Toth, S. Zörner:
„Leichtgewichtige Software-
Reviews mit LASR“, Informatik
Aktuell 2024



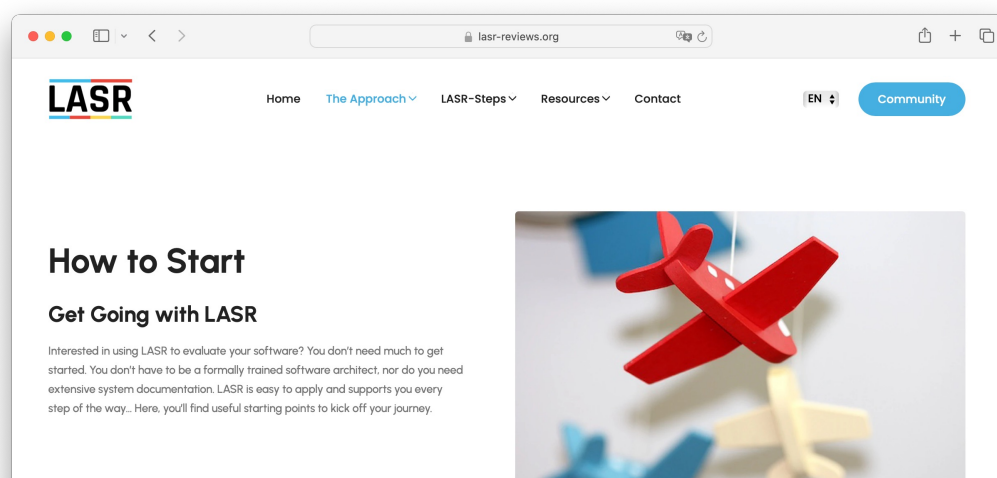
05.

Weitere Informationen

Neugierig geworden? Es gibt noch mehr zu entdecken!



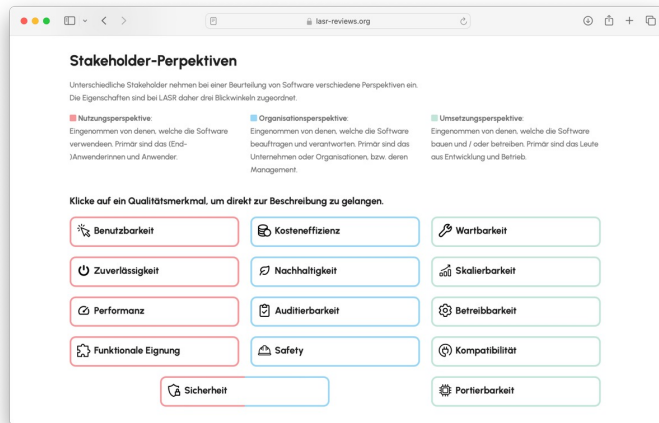
„Offizielle“ LASR-Webseite (EN + DE)



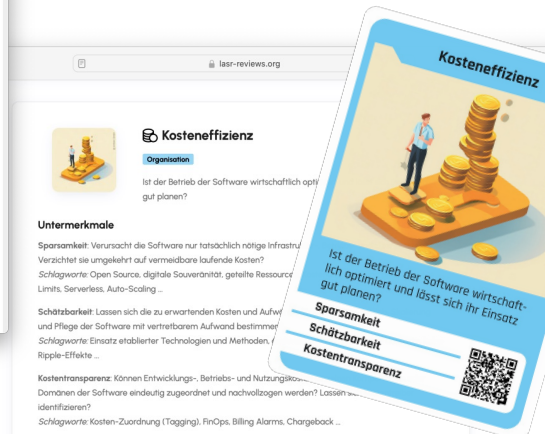
→ lasr-reviews.org

LASR-Qualitätsmodell

Beschreibungen der in Top-5-Challenger eingesetzten 14 Qualitätsmerkmale online.



→ lasr-reviews.org/quality-model/



IEEE-Paper

S. Toth, S. Zörner:
**„Team-Driven Architecture
 Evaluation with LASR“**
 IEEE 23rd International
 Conference on Software
 Architecture, Amsterdam 2026



→ lasr-reviews.org/research/lasr-paper-ieee-icsa-2026/

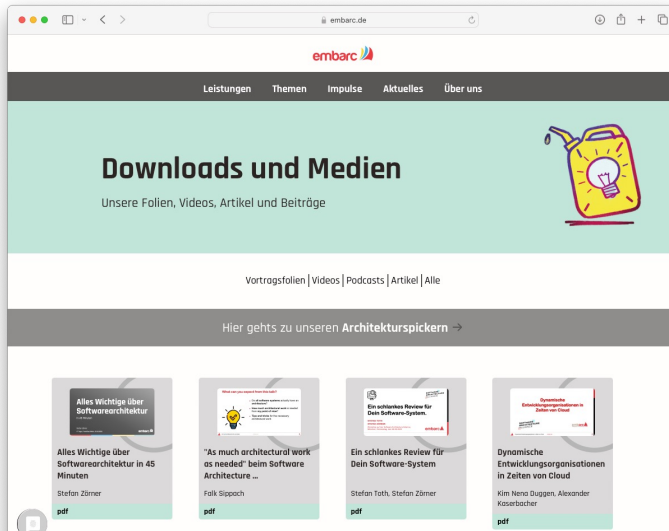


embarc.de

LASR-Methode für Software-Reviews

50

Folien als PDF zum Download



→ embarc.de/download/



Unser Line-Up

Vorträge, Workshops, neue Perspektiven und Plaudern
in lockerer Punsch-Atmosphäre am 7. Dezember 2026!



Ulrich Lehner
(LVM Versicherungen)



Claudia Fröhling
(socreatory)



Tom Asel
(tangible concepts)



Roman Hecht
(Continental Reifen)



Niklas Trentmann



Falk Sippach



Stefan Zörner



Michael Rabe
(Continental Reifen)



Lisa Maria Schäfer
(socreatory)



Stefan Toth



Alle Infos & Anmeldung: embarc.de/architektur-punsch/





Vielen Dank.

Gerne Verlinken auf LinkedIn ...

 [linkedin.com/in/stefan-zoerner](https://www.linkedin.com/in/stefan-zoerner)



embarc 

