# Aufgaben von Softwarearchitekten

Softwarearchitekten müssen deutlich mehr leisten, als „nur“ Softwarearchitekturen zu entwerfen: Sie sind „Anwalte der Auftraggeber“ und Berater für Manager und für das Realisierungsteam.

## Architekten konstruieren, entwerfen und implementieren

Softwarearchitekten (gemeinsam mit Entwicklungsteams) entwerfen und konstruieren alle Bestandteile, die für Entwicklung, Betrieb und Wartung eines Softwaresystems notwendig sind:

* **Bausteine**: Architekten und Entwickler konstruieren Systeme aus Bausteinen, die ihrerseits wieder komplexe Subsysteme sein können. Bausteine besitzen klare *Verantwortlichkeiten*.
* **Schnittstellen**: über Schnittstellen kommuniziert ein System mit der Außenwelt. Schnittstellenbilden die Grundlage der „Vertrage“, auf deren Basis die Bausteine miteinander arbeiten (*design by contract*). Erst die Zusammenarbeit der Bausteine über Schnittstellen ermöglicht es dem System, seine Aufgaben zu erfüllen.
* **Strukturen**: Durch Bausteine und deren Zusammenwirken entwerfen Architekten sowohl statische als auch dynamische Strukturen.

Ich bin der Meinung, dass Architekten ihre Entwurfs-, Struktur- und Technologieentscheidungen am besten in einem (kleinen) Team treffen – nur in Ausnahmefallen allein.

## Architekten entscheiden

* „… schnelle Folge suboptimaler Entwurfsentscheidungen“: Architekten müssen viele Enscheidungen treffen: Welche Bausteine, welche Schnittstellen, welche Ablaufe? Welche technischen Frameworks? Selbst implementieren, kaufen oder einen Mittelweg davon? Welches Teil Team entwickelt welche Komponenten? Wie sollen die Bausteine der Architektur heißen?
* „… *die meist im Dunkeln getroffen werden*“: Architekten wissen oftmals wenig über die langfristigen Konsequenzen ihrer Entscheidungen. Teilweise zeigt sich erst Monate (oder Jahre!) später, ob eine Architekturentscheidung vernünftig und angemessen war. Darüber hinaus arbeiten Softwarearchitekten oftmals mit innovativen technischen Frameworks, Betriebssystemen oder sonstigen Dingen, deren genaues Verhalten sie gar nicht kennen können. In solcher Unsicherheit hilft iteratives Vorgehen weiter. Entscheidungen großer Tragweite sollten Sie angemessen dokumentieren.

## Architekten garantieren die Erfüllung von Anforderungen

Softwarearchitekten stellen die *Machbarkeit* und *Erfüllung* von Anforderungen sicher. Sie gewährleisten, dass Anforderungen einerseits erfüllbar sind und andererseits auch erfüllt werden. Dies bezieht sich sowohl auf die funktionalen Anforderungen (*required capabilities)* als auch auf die nichtfunktionalen Anforderungen und Randbedingungen (*required constraints*). Architekten belegen die Machbarkeit des Systems durch Prototypen. Nicht zuletzt sorgen Architekten auch dafür, dass Systeme mit angemessenen Kosten realisiert werden können!

## Architekten beraten

Softwarearchitekten beraten andere Projektbeteiligte in architekturrelevanten Fragestellungen. Sie beraten:

* Management und Auftraggeber bei der Projektplanung und -organisation.
* Auftraggeber und Analyseteams hinsichtlich der Machbarkeit von Anforderungen. Dazu unterstützen sie bei der Bewertung von Kosten und Nutzen von Anforderungen. Sie klären die Auswirkungen von Anforderungen auf die Struktur, die Realisierung und den Betrieb von Systemen.
* Projektleiter beim Management (technischer) Risiken. Architekten sollten zusätzlich auch organisatorische Risiken kennen und berücksichtigen. Weiterhin helfen sie Projektleitern bei der Organisation und Steuerung des Implementierungsteams.
* Implementierungsteams bezüglich der Umsetzung der Architektur in Software. Dazu müssen Architekten das Team von der Architektur überzeugen und bei Bedarf ausbilden und unterstützen.
* Hardwarearchitekten und Betreiber des Systems hinsichtlich der Anforderungen, die das System an die zugrunde liegende Hardware stellt.
* Qualitätssicherung und Test hinsichtlich der Kritikalität und Testbarkeit von Systembestandteilen.

## Architekten dokumentieren

Damit Projekte agil, flexibel und kurzfristig wandlungsfähig bleiben, müssen Architekten in hohem Maße angemessen und bedarfsgerecht arbeiten.

Im Wesentlichen kommt es auf angemessene Dokumentation und Kommunikation an. Sowohl Art als auch Umfang und Detaillierung müssen sich an den Bedürfnissen der jeweiligen Adressaten orientieren.

Manchmal genügt eine kurze Skizze („auf der Ruckseite eines gebrauchten Briefumschlags“), manchmal ein detailliertes UML-Diagramm, und in anderen Fällen treffen umfangreiche Dokumente die Bedürfnisse der Adressaten.

## Architekten sind Diplomaten und Akrobaten

Als Diplomaten schließen Architekten Kompromisse zwischen widersprüchlichen oder konkurrierenden Forderungen.

Als Akrobaten balancieren Architekten mit einer Vielzahl von Faktoren, die sich gegenseitig beeinflussen. Die untere Tabelle zeigt einige solcher Faktoren (nach [Rechtin2000]), die im Wettbewerb miteinander stehen (und für Architekten die Arbeit interessant machen). Das Bild mit dem Gummiband zeigt eine Lösung in Abhängigkeit von der Stärke solcher Forderungen und Einflüsse.

|  |  |  |
| --- | --- | --- |
| Form | 🡸🡺 | Funktion |
| Strikte Kontrolle | 🡸🡺 | Agile Entscheidungen |
| Kosten & Termine | 🡸🡺 | Leistung & Qualität |
| Komplexität | 🡸🡺 | Einfachheit bzw. Flexibilität |
| Technische Innovation | 🡸🡺 | Etablierte Technologien |
| Top-Down-Planung | 🡸🡺 | Bottom-Up-Realisierung |
| Kontinuierliche Verbesserung | 🡸🡺 | Produktstabilität |
| Minimale Schnittstellen | 🡸🡺 | Enge Integration |

Abbildung Konkurrierende Faktoren, die Kompromisse und Balance benötigen

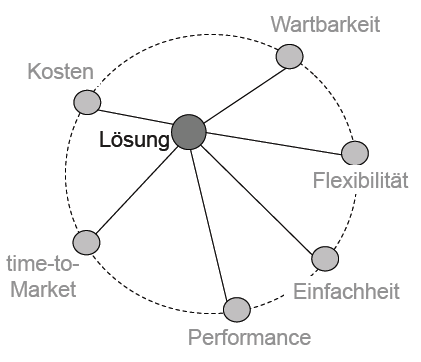


Abbildung Gummiband Lösung als Kompromiss

## Architekten vereinfachen

Eine wichtige Regel für Architekten lautet: Vereinfache! Vereinfache! Vereinfache! Eine andere Lesart:

Weniger ist (manchmal) mehr!

Einfache Strukturen sind leichter und günstiger realisierbar, einfacher verständlich, weniger Fehler anfällig. Die zuverlässigste, preiswerteste und robusteste Komponente eines Systems ist diejenige, die erst gar nicht realisiert werden muss!

Einfacher bedeutet manchmal „kleiner“ – dabei konnte die Einführung von Microservices helfen. Allerdings benötigt dieser Architekturansatz erhebliche organisatorische Unterstützung, beispielsweise müssen Sie dazu cross-funktionale Teams einführen.

## Architekturen kommunizieren

Architekten müssen den anderen Stakeholdern die Architektur vermitteln und sie von den Architekturentscheidungen überzeugen. Dazu gehören sowohl der Wille als auch die Fähigkeit, technische Sachverhalte für unterschiedliche Stakeholder angemessen aufzubereiten, zu präsentieren und zu diskutieren.

Gute Softwarearchitekten zeichnen sich dadurch aus, dass sie ihre Ideen, Entwurfe und Entscheidungen aktiv an die übrigen Projektbeteiligten kommunizieren können.

* Architekten erläutern und argumentieren ihre Entscheidungen. Wenn nötig, verteidigen sie diese auch gegen Angriffe. Hierfür benötigen sie diplomatisches Geschick.
* Architekten überzeugen das Entwicklungsteam von Strukturen und Schnittstellen, indem sie Vor- und Nachteile transparent darstellen und unter den Rahmenbedingungen und Einflüssen des konkreten Systems gegeneinander abwägen.
* Architekten präsentieren und vermarkten die Architektur, sodass möglichst alle Beteiligten sie als DIE Architektur akzeptieren und in technischer Hinsicht an einem Strang ziehen.
* Sie unterrichten und coachen: treten als Berater, Trainer oder Lehrer auf, um das benötigte Know-how im Team zu verbreiten.

Für diese Aufgaben benötigen Softwarearchitekten starke kommunikative Fähigkeiten, sollten motivieren, präsentieren und argumentieren können. Fundierte technische Kenntnisse stellen lediglich eine *notwendige* Voraussetzung für diesen Teil der Architekturarbeit dar.

## Architekten bewerten

Architekten müssen die Gute der Architekturen bewerten, um jederzeit den Grad der Zielerreichung zu kennen. Sie müssen wissen, ob und an welchen Stellen der Systeme nichtfunktionale Anforderungen (wie etwa Performance) riskant oder kritisch sind. Aus dieser objektiven Bewertung heraus können Architekten Maßnahmen zur Optimierung oder Risikominderung ableiten – in der Regel gemeinsam mit anderen Verantwortlichen.

## Architekten explizieren

Architekten sollten wichtige Annahmen und Voraussetzungen explizit machen, um die möglichen Probleme *impliziter Annahmen* zu adressieren. Sie vermeiden dadurch mögliche Missverständnisse zwischen Beteiligten.

## Architekten brauchen Mut

Am Ende eines erfolgreichen Projekts ist es einfach, Entwurfsentscheidungen zu kritisieren. Zu diesem Zeitpunkt wissen alle Beteiligten über Technologien, Produkte und auch Anforderungen genau Bescheid.

Architekten mitten im Stress der Entwicklung können solche Dinge teilweise nur vermuten. Architekten verfügen aus Zeitgründen oftmals nicht über genügend Informationen, um optimale Entscheidungen zu treffen. Damit die Entwicklung weiterlaufen kann, müssen Architekten (und Entwicklungsteams) in solchen Fällen Mut zu (möglicherweise) suboptimalen Entscheidungen unter Unsicherheit aufbringen.

Architekten können die Schuld nicht auf diejenigen schieben, die die Anforderungen gestellt haben. Allerdings sollten Architekten die übrigen Beteiligten möglichst frühzeitig auf mögliche Konsequenzen bestimmter (kritischer) Anforderungen hinweisen (und damit das Risikomanagement ihrer Projektleiter unterstützen!).

## Zusammenfassender Überblick

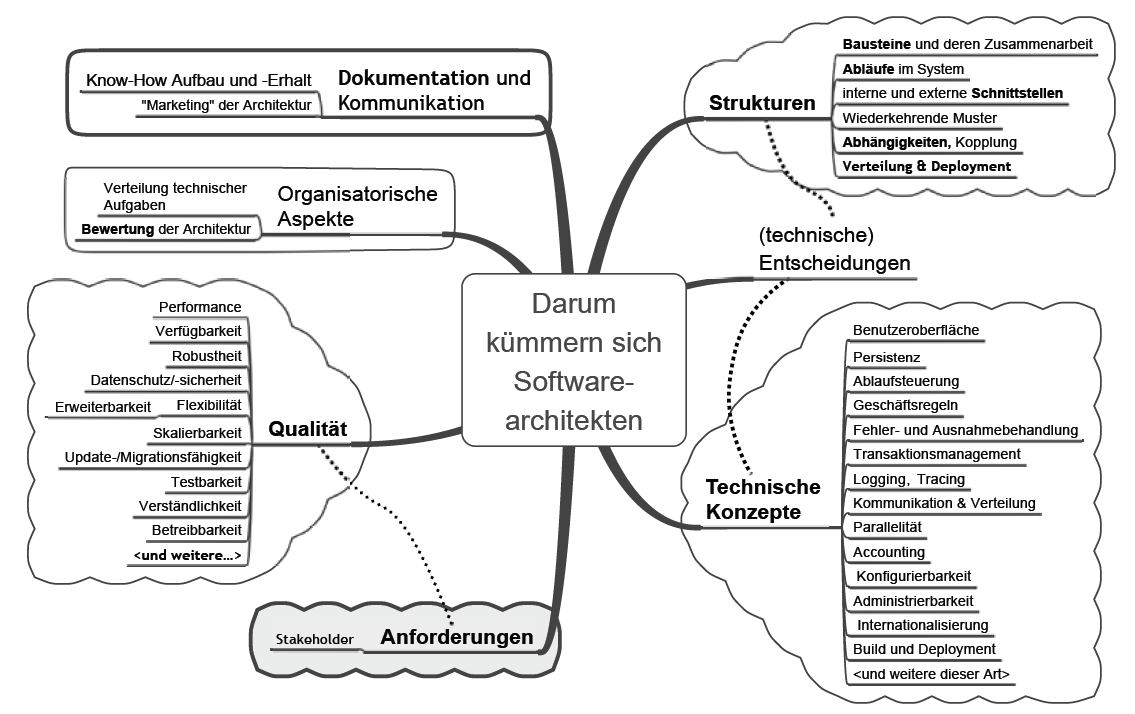


Abbildung Aufgaben von Softwarearchitekten