

Nur weil ich einen Hammer dabei habe, ist nicht jedes Problem ein Nagel!

Architektur Definitionen

- Kent Beck

Software architecture is what software architects do.

- Martin Fowler (<http://martinfowler.com/articles/designDead.html>)

To me the term architecture conveys a notion of the core elements of the system, the pieces that are difficult to change. A foundation on which the rest must be built.

- Eoin Woods

Software architecture is the set of design decisions which, if made incorrectly, may cause your project to be cancelled.

- Carnegie Mellon University (<http://www.sei.cmu.edu/architecture/start/glossary/index.cfm>)

abstraction assignment behavior cases
components comprise computation connections
consists constraints data defined definition described
description design details different documenting
elements form functionality governing
implementation importance include information interact
interfaces level models modules organization performance process
programs properties relations relationships
requirements software specific
structure style system together
type units used view

Design und Randbedingungen

Gegeben sei ein Problemraum. Aus diesem ergeben sich unsere Anforderungen. Für die Anforderungen suchen wir eine Lösung. Dafür versuchen wir, in einem riesigen Lösungsraum eine konkrete Lösung herauszuarbeiten. Eine Lösung besteht dabei aus einer Summe von unterschiedlich großen Design- oder Implementierungs-Entscheidungen.

In einem typischen Projekt ist der Lösungsraum zunächst durch Randbedingungen eingeschränkt. Diese können nur in den seltensten Fällen verändert werden. Sie sind hoffentlich sinnvoll gewählt, meist aber auf einen Unternehmensfokus ausgerichtet und nicht auf dieses konkrete Projekt.

Die Randbedingungen sind aber meist nicht die einzige Einschränkung des Lösungsraumes im Projekt. Die zweite große Einschränkung machen wir (die Projektverantwortlichen) selbst durch unsere Architektur. Diese wählen wir sinnvoll und angemessen für das Projekt und wollen dadurch erreichen, dass der verbleibende Lösungsraum nun so gestaltet ist, dass wir schneller zu besseren Lösungen kommen.

Wir suchen im durch Randbedingungen und Architektur verkleinerten Lösungsraum aller Design Entscheidungen nach einer konkreten Lösung. Die Architektur unterstützt diese Suche indem sie den Lösungsraum so verkleinert, dass wir schneller bessere Lösungen finden können.

Last Responsible Moment

- Es geht um Entscheidungen mit einem gewissen Risiko.
- Die Chance eine gute Entscheidung zu treffen ist höher wenn mehr Informationen vorliegen.
- Je weiter das Projekt vorangeschritten ist, desto mehr Informationen liegen vor.
- Andererseits hat jede Entscheidung eine Dringlichkeit.
- Sie ist dringender je größer die Wahrscheinlichkeit ist, sie dann implizit zu treffen da man ansonsten nicht weiter arbeiten kann.
- Implizite Entscheidungen bei riskanten Problemen wollen wir vermeiden.

Der Last Responsible Moment ist der Zeitpunkt für eine Entscheidung, der so spät wie möglich aber so früh wie nötig ist.

Software Architektur ist unausweichlich

- Emergent ja, aber das alleine reicht nicht
- Bewusstes Entscheiden unterstützt, ein definiertes Ziel zu erreichen
- Man kann sich auch bewusst entscheiden, nicht zu entscheiden – die Konsequenzen sollte man aber dabei bedacht haben

Software Architektur ist kein Selbstzweck

- Architektur um der Architektur Willen ist Waste
- Ziel von Architektur ist, Projektrisiken und damit letztendlich Projektkosten zu kontrollieren
- Architektur hat einen mittel- bis langfristigen Horizont und will
 - Entscheidungsfindung optimieren (sowohl bei der Dauer als auch beim Ergebnis)
 - Entwicklung neuer Features unterstützen
 - Ausbessern von Fehlern erleichtern

Die Person „Architekt“ ist nicht zwingend nötig – die Rolle schon

- Rolle hat viele Aspekte in den Bereichen Soft Skills, Methodik und Technik, unter Anderem
 - Kommunikation, Dokumentation, Koordination, Verstehen, Klären, Hinterfragen, Identifizieren, Entscheiden, Managen, Entwickeln, Regeln aufstellen, Reflektieren, Bewerten, Feedback einholen, Überarbeiten, Verifizieren, Integrieren, Vermitteln, Sicherstellen
- Rolle kann auf das Team verteilt sein
- Falls nicht macht es Sinn, den Architekten im Team mitarbeiten zu lassen
 - Vermeidung von Elfenbeinturmarchitektur
 - Vermeidung des „Not-Invented-Here“ Syndrom

Grundtechniken – Entscheidungsfindung

- Was ist das Problem?
- Welche Einflussfaktoren gibt es?
- Welche Annahmen kann ich treffen?
- Welche Alternativen gibt es?
- Wie lautet die Entscheidung?

Grundtechniken – Meta Architektur

- Architekturziele
 - Zum Beispiel „Unsere Architektur erlaubt es uns, die Oberflächentechnologie alle 5 Jahre mit möglichst geringem Aufwand auszutauschen.“
- Architekturprinzipien
 - Zum Beispiel „Abhängigkeiten gehen immer in Richtung des stabilen Elements“
- Architekturmuster
 - Zum Beispiel MVC

Grundtechniken – Dokumentation

- Schreibe aus Sicht des Lesers
- Keine unnötigen Wiederholungen
- Nutze Standardstrukturen
- Halte Begründungen fest
- Prüfe die Tauglichkeit indem Du Feedback einholst

Weiterführende Themen

- Qualitätskriterien
- Cross Cutting Concerns
- Einbindung in den Prozess
- Architekturbewertung
- Soft Skills für Architekten
- Software Design
- Werkzeuge für Software Architektur
- Enterprise Architektur, Service Oriented Architecture, ...

Wer mehr wissen möchte

- Bücher
 - Martin Fowler - Patterns of Enterprise Application Architecture
 - Gernot Starke - Effektive Software Architekturen
 - Erich Gamma et al - Design Patterns
 - Eric Evans – Domain Driven Design
- Internet
 - <http://www.sei.cmu.edu/architecture/start/glossary/>
 - <http://www.iso-architecture.org/ieee-1471/>
 - <http://agilemanifesto.org>
 - <https://leanpub.com/software-architecture-for-developers/read>
 - <http://architecturalkatas.site44.com>
 - <http://ccd-school.de/coding-doj/>
 - <http://aim42.org>
- Schulungsprogramme
 - iSAQB Certified Professional Software Architect (CPSA)
 - <http://www.isaqb.org/certifications/>
 - Open Group CA
 - <http://opengroup.org/certifications/professional/open-ca>
- Neugierig bleiben
- Entscheidungen bewusst treffen
 - Oder bewusst nicht treffen