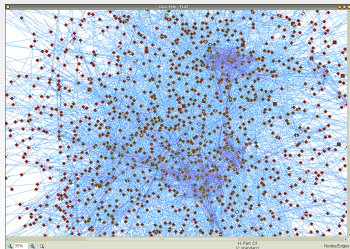


Architektursichten und -blickwinkel (IEEE P1471 2002)

Definition

Architektursicht (View):

Repräsentation eines ganzen Systems aus der Perspektive einer kohärenten Menge von Anliegen.



Definition

Architekturblickwinkel (Viewpoint):

Spezifikation der Regeln und Konventionen, um eine Architektursicht zu konstruieren und zu benutzen.

Ein Blickwinkel ist ein Muster oder eine Vorlage, von der aus individuelle Sichten entwickelt werden können, durch Festlegung von

- Zweck,
- adressierte Betrachter,
- und Techniken für Erstellung, Gebrauch und Analyse.

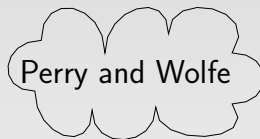
calls

function

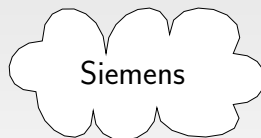
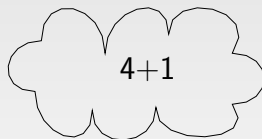
- **Konzeptioneller Blickwinkel:** beschreibt logische Struktur des Systems; abstrahiert weitgehend von technologischen Details
- **Modulblickwinkel:** beschreibt die statische logische Struktur des Systems
- **Ausführungsblickwinkel:** beschreibt die dynamische logische Struktur des Systems
- **Code-Blickwinkel:** beschreibt die „anfassbaren“ Elemente des Systems (Quelldateien, Bibliotheken, ausführbare Dateien etc.)



Verbreitete Blickwinkel




Verbreitete Blickwinkel



Verbreitete Blickwinkel



Zachman



Perry and Wolfe



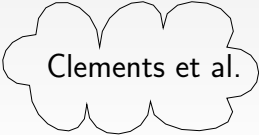
4+1



Siemens



...

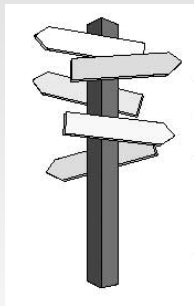


Clements et al.

Blickwinkelkategorisierung (Clements u. a. 2002)

- **M:** module
 - decomposition
 - use
 - generalization
 - layers
- **CC:** component & connectors
 - pipe and filter
 - shared data
 - publish and subscribe
 - client server
 - peer-to-peer
 - communicating processes
- **A:** allocation
 - deployment
 - implementation
 - work assignment

- ① Einflussfaktoren identifizieren
 - Produktfunktionen und -attribute
 - technologische Faktoren
 - organisatorische Faktoren
- ② konkurrierende Faktoren feststellen
- ③ Kompromisse für Faktorenkonflikte durch Strategien finden
- ④ iterativer Entwurf der verschiedenen Sichten



Teil II

Qualität von Software-Architekturen



Funktionalität versus Qualität

Definition

Funktionalität: Umsetzung der funktionalen Anforderungen; die Fähigkeit eines Software-Systems, auf eine Eingabe die erwartete Ausgabe zu produzieren.

Funktionalität versus Qualität

Definition

Funktionalität: Umsetzung der funktionalen Anforderungen; die Fähigkeit eines Software-Systems, auf eine Eingabe die erwartete Ausgabe zu produzieren.

Funktionalität kann durch beliebige Strukturen umgesetzt werden; ist damit weitgehend unabhängig von Struktur.

Software-Architektur schränkt mögliche Strukturen ein aufgrund anderer Qualitätsattribute.

Funktionalität versus Qualität

Definition

Funktionalität: Umsetzung der funktionalen Anforderungen; die Fähigkeit eines Software-Systems, auf eine Eingabe die erwartete Ausgabe zu produzieren.

Funktionalität kann durch beliebige Strukturen umgesetzt werden; ist damit weitgehend unabhängig von Struktur.

Software-Architektur schränkt mögliche Strukturen ein aufgrund anderer Qualitätsattribute.

Definition

Qualität ist der Grad, in dem ein Satz inhärenter Merkmale Anforderungen erfüllt.

EN ISO 9000:2005

Beispiele für Software-Qualitätsaspekte

- Änderbarkeit
- Testbarkeit
- Sicherheit
- Robustheit
- Gebrauchstauglichkeit (Usability)
- Performanz
- Verfügbarkeit
- Skalierbarkeit
- Portierbarkeit
- ...

- Architektur ist kritisch für die Realisierung vieler Qualitäten
 - Qualitäten müssen durch Konstruktion eingebaut werden
 - Qualitäten können und sollen auf Architekturebene evaluiert werden

- Architektur ist kritisch für die Realisierung vieler Qualitäten
 - Qualitäten müssen durch Konstruktion eingebaut werden
 - Qualitäten können und sollen auf Architekturebene evaluiert werden
- Architektur alleine genügt nicht zur Erreichung der Qualitäten
 - Architektur bildet nur die Grundlage
 - Implementierungsdetails sind maßgebend

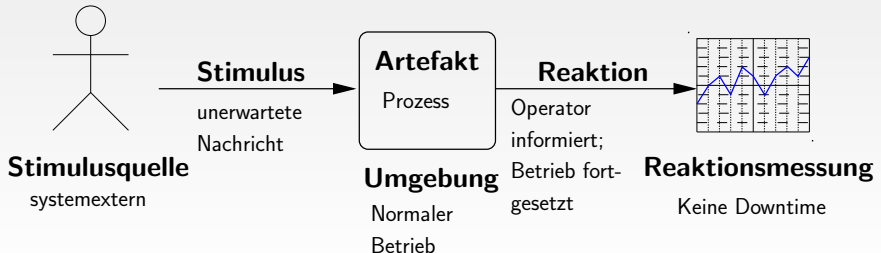
- Architektur ist kritisch für die Realisierung vieler Qualitäten
 - Qualitäten müssen durch Konstruktion eingebaut werden
 - Qualitäten können und sollen auf Architekturebene evaluiert werden
- Architektur alleine genügt nicht zur Erreichung der Qualitäten
 - Architektur bildet nur die Grundlage
 - Implementierungsdetails sind maßgebend
- Qualitätsattribute können im Konflikt zueinander stehen; Architektur ist ein Kompromiss

- Architektur ist kritisch für die Realisierung vieler Qualitäten
 - Qualitäten müssen durch Konstruktion eingebaut werden
 - Qualitäten können und sollen auf Architekturebene evaluiert werden
- Architektur alleine genügt nicht zur Erreichung der Qualitäten
 - Architektur bildet nur die Grundlage
 - Implementierungsdetails sind maßgebend
- Qualitätsattribute können im Konflikt zueinander stehen; Architektur ist ein Kompromiss
- Qualitätsattribute müssen objektiv und operational beschrieben sein
 - konkrete messbare Szenarien

Definition

Qualitätsattributsszenario ist eine operationale Anforderung hinsichtlich eines Qualitätsattributs (Bass u. a. 2003):

- wenn ein bestimmtes Ereignis eintritt (Stimulus)
- in einer bestimmten Situation (Umgebung),
- das von einem bestimmten Auslöser kommt (Stimulusquelle)
- und auf einen bestimmten Gegenstand einwirkt (Artefakt),
- dann soll eine geforderte Reaktion
- in einer messbaren Art eintreten (Reaktionsmessung).



Kategorien von Software-Architektur-Qualitäten

- Systemqualitäten (Verfügbarkeit, Änderbarkeit, Performanz, Sicherheit, Testbarkeit, Gebrauchstauglichkeit etc.)
- Geschäftsqualitäten, z.B. Time-To-Market
- Qualitäten der Architektur selbst, z.B. konzeptionelle Integrität, die indirekt die anderen Qualitäten beeinflussen

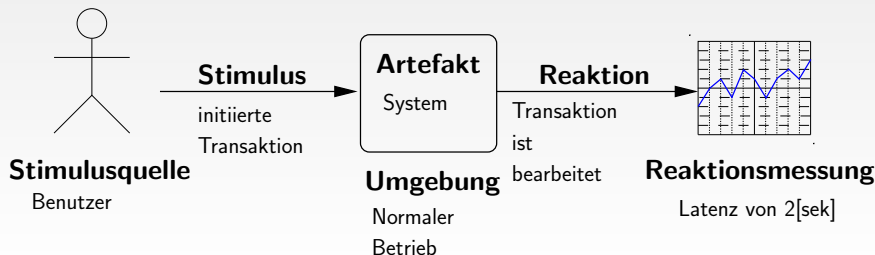
Allgemeine Szenarien:

Quelle	intern/extern
Stimulus	periodische, sporadisch, stochastische Ereignisse
Artefakt	System
Umgebung	Normalbetrieb, ausgelastet
Reaktion	Bearbeitung von Stimuli, Änderung von Service-Levels
Maß	Latenz, Deadline, Durchsatz, Versatz, Versäumnisrate, Datenverlust

Allgemeine Szenarien:

Quelle	intern/extern
Stimulus	periodische, sporadisch, stochastische Ereignisse
Artefakt	System
Umgebung	Normalbetrieb, ausgelastet
Reaktion	Bearbeitung von Stimuli, Änderung von Service-Levels
Maß	Latenz, Deadline, Durchsatz, Versatz, Versäumnisrate, Datenverlust

Spezielles Szenario:



Allgemeine Szenarien:

Quelle	Endbenutzer, Entwickler, Systemadministrator
Stimulus	Wunsch, Funktionalität hinzuzufügen, zu entfernen, abzuändern, zu variieren bzw. Qualitätsaspekt zu verändern
Artefakt	System (Benutzeroberfläche, Plattform, Umgebung, kooperierendes System)
Umgebung	Laufzeit, Ladezeit, Übersetzungszeit, Entwurfszeit
Reaktion	Lokalisierung, Änderung, Test, Auslieferung der Architekturkomponenten
Maß	Kosten in Form von Anzahl der betroffenen Komponenten, Aufwand, Geld; Ausmaß des Einflusses auf andere Qualitätsattribute

Spezielles Szenario:

