

Aufgaben von Softwarearchitekten

1. Anforderungen und Randbedingungen klären und hinterfragen
2. Strukturentscheidungen hinsichtlich Systemzerlegung und Baueinstruktur treffen
3. Übergreifende technische Konzepte entscheiden und bei Bedarf umsetzen
4. Softwarearchitektur kommunizieren und dokumentieren
5. Umsetzung und Implementierung der Architektur überwachen
6. Architektur bewerten (insbesondere Risiken bzgl. Qualitätsmerkmale)

Schnittstellen-Definition

1. Wohldefinierter Zugangspunkt zum System oder dessen Bausteinen
 2. beschreibt die Eigenschaften: Attribute, Daten und Funktionen
- Ziel:** möglichst präzise Definition

Arten von Schnittstellen

1. Standardschnittstelle: von **außerhalb** definiert (standardisiert)
2. Angebotene Schnittstelle: definiert durch den Baustein der S. **exportiert**
3. Angeforderte Schnittstelle: definiert durch den Baustein der S. **importiert**
4. Unabhängige Schnittstelle: jeweils eigene (**inkompatible**) S. → **Adapter**-Pattern

Architekturmuster

- POSA (Buschmann et al.): Layers (Schichtenarchitektur), Pipes and Filters, Blackboard, Mikrokern
- PoEAA (Fowler, Patterns of Enterprise Application Architecture): Gateway, Registry, Service Layer
- EIP (Hohpe, Enterprise Integration Patterns): Messaging, Async-Pattern
- MVC (Model-View-Controller): Benachrichtigung der View über Datenänderung mit **Observer**-Pattern

Entwurfsmuster (Design Pattern)

- Erzeugungsmuster (Creational): Singleton, Abstract Factory, Factory Method usw.
- Strukturmuster (Structural): Adapter, Facade, Composite usw.
- Verhaltensmuster (Behavioral): Observer, Command, Strategy usw.

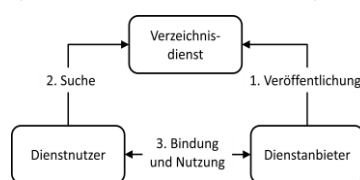
Qualitätsmerkmale nach ISO 9126 (→ Produktqualität)

1. **Funktionalität:** Angemessenheit, Richtigkeit, Interoperabilität, Sicherheit, Ordnungsmässigkeit
2. **Zuverlässigkeit:** Reife, Fehlertoleranz, Wiederherstellbarkeit
3. **Benutzbarkeit:** Verständlichkeit, Erlernbarkeit, Bedienbarkeit
4. **Effizienz:** Zeitverhalten, Verbrauchsverhalten
5. **Änderbarkeit:** Analysierbarkeit, Modifizierbarkeit, Stabilität, Prüfbarkeit
6. **Übertragbarkeit:** Anpassbarkeit, Installierbarkeit, Konformität, Austauschbarkeit

Entwurfsprinzipien

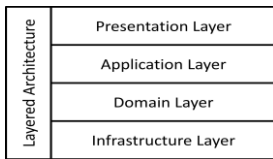
- Geheimnisprinzip
- Lose Kopplung / Hohe Kohäsion
- Separation of Concern
- Offen-Geschlossen-Prinzip
- Liskovsches Substitutionsprinzip
- Umkehr von Abhängigkeiten
- Dependency Injection

SOA (Service Oriented Architecture)



DDD (Domain Driven Design)

- Entities
- Value Objects
- Aggregates
- Factories
- Repositories
- Services



Werkzeuge für Softwarearchitekten

- Modellierungswerkzeuge (UML)
- W. zur statischen & dynamischen Analyse
- Generierungswerkzeuge (MDA)
- Anforderungsmanagementwerkzeuge
- Dokumentationswerkzeuge (Wiki)
- Build- und Konfigurationsmanagement

MDA (Model Driven Architecture)

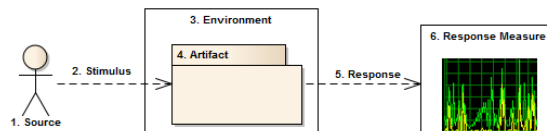
- CIM (Computation Independent Model): umgangssprachliche Beschreibung
 - PIM (Platform Independent Model): plattformunabhängiges Modell für Geschäftsprozesse
 - PSM (Platform Specific Model): plattformabhängiges Modell für Architektur, Services
 - Codemodell: Zielplattform
- Zentrale Begriffe:** Modelltransformation, Codegenerierung, UML-Profil

ATAM Prozess (Phasen & Prozessschritte) → Szenariobasiertes Verfahren

- a) Präsentation (1. Bewertungsmethode, 2. Geschäfts- & Architekturziele, 3. Systemarchitektur)
- b) Recherche und Analyse (4. Architekturansatz identifizieren, 5. Qualitätsbäume und Szenarien erstellen, 6. Architekturansatz analysieren (Architektur-zentriert))
- c) Test (7. Szenarien-Brainstorming und -Priorisierung, 8. Architekturansatz analysieren (Stakeholder-zentriert))
- d) Reporting (9. Ergebnisse präsentieren)

Qualitätsszenarien

- Nutzungsszenarien
- Änderungsszenarien
- Stress- oder Grenzszenarien



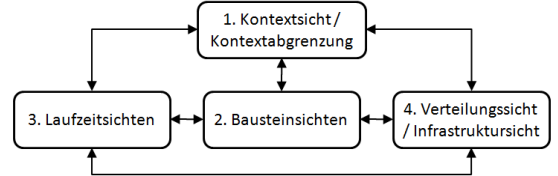
Tätigkeiten beim Architekturf Entwurf

1. Anforderungen und Randbedingungen analysieren
2. Architektursichten und technische Konzepte entwerfen
3. Architektur- und Entwurfsentscheidungen treffen
4. Umsetzung begleiten und prüfen

Bausteine

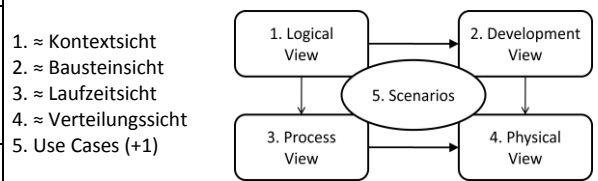
1. Export und Import von Schnittstellen
2. Kapselung und Austauschbarkeit
3. Konfiguration und hierarchische (De-)Kompensation

Architektursichten

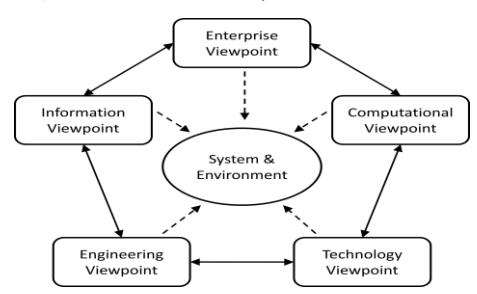


1. Paket-, Komponenten-, Verteilungsdiagramme
2. Komponenten- und Klassendiagramme
3. Sequenz- und Aktivitätsdiagramme
4. Verteilungsdiagramme

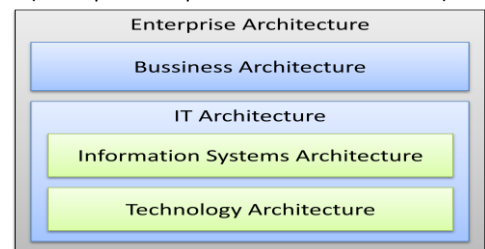
Kruchten, 4+1 Sichtenmodell (RUP – Rational Unified Process)



RM-ODP (Reference Model of Open Distributed Processing)



TOGAF (The Open Group Architecture Framework)



Weitere Frameworks zur **Dokumentation von Architekturen**

- IEEE-1471: Architectural Description of Software-Intensive Systems
- SAGA: Standards und Architekturen für eGovernment-Anwendungen
- FMC: Fundamental Modeling Concepts (SAP)
- arc42: Freies Template zur Beschreibung von SW-Architekturen
- Zachman EA Framework

ATAM Qualitätsbäume (utility tree)

- Szenarien befinden sich in den Blättern des Qualitätsbaums
- Stakeholder priorisieren Szenarien nach geschäftlichem Nutzen

