

A GOOD ARCHITECT LEAVES A FOOTPRINT



Langlebige Softwarearchitekturen – der Weg aus den technischen Schulden

Dr. Carola Lilienthal, cl@wps.de, [@cairolali](https://www.instagram.com/cairolali)
www.wps.de

WPS - Workplace Solutions GmbH Hans-Henny-Jahnn-Weg 29 22085 HAMBURG





**Business-Software,
die Spaß macht!**

Software- Architektur	Anforderungs- ermittlung	Leitstand und Interaktion	Individual- software
			

WPS – Workplace Solutions GmbH



WPS WORKPLACE SOLUTIONS

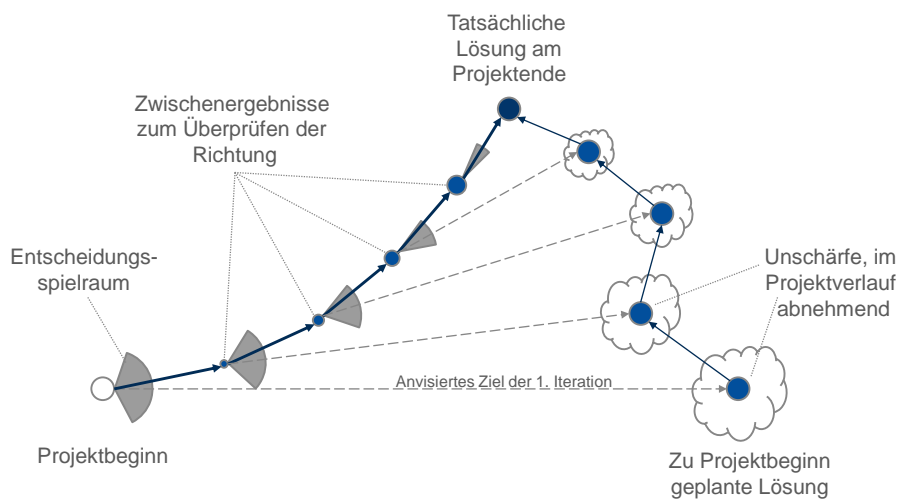
Diplom-Informatikerin
Dr. Carola Lilienthal
Geschäftsführerin

WPS - Workplace Solutions GmbH
Hans-Henny-Jahns-Weg 29
22085 Hamburg
Telefon +49 40 229 499-0
Telefax +49 40 229 499-299
Mobil +49 170 184 77 11
cl@wps.de
www.wps.de

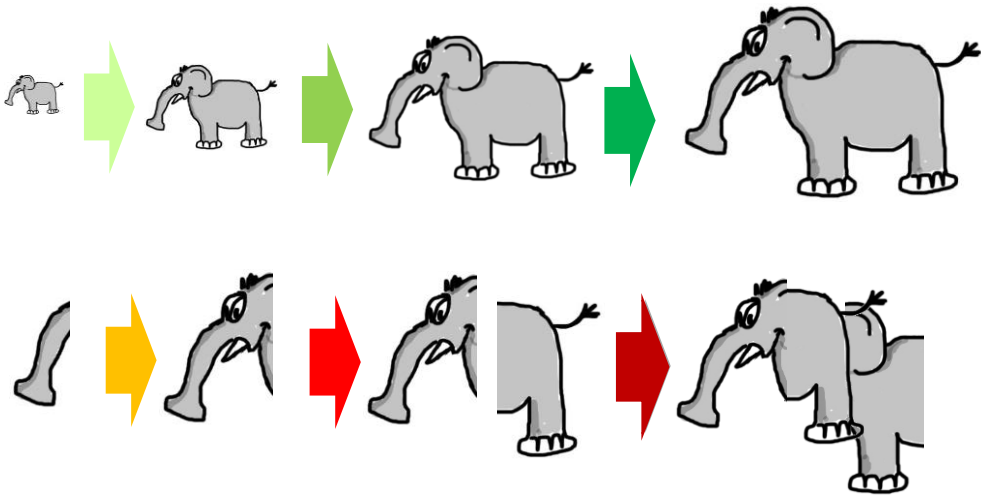
@cairolali

@cairolali
cl@wps.de
www.lisa.de

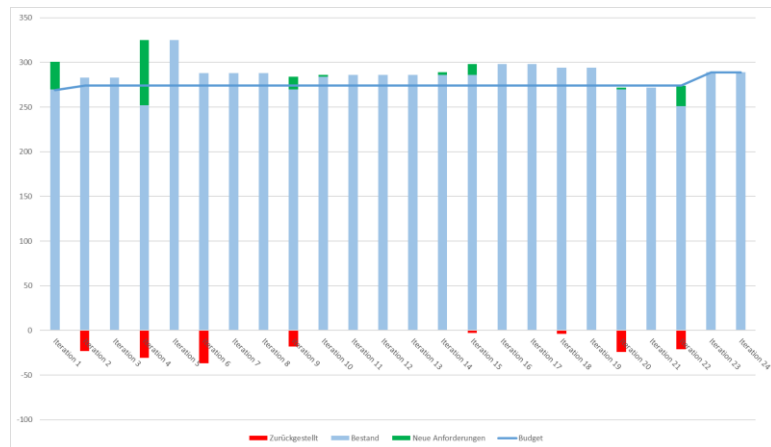
Software-Entwicklung/Einführung = Expedition



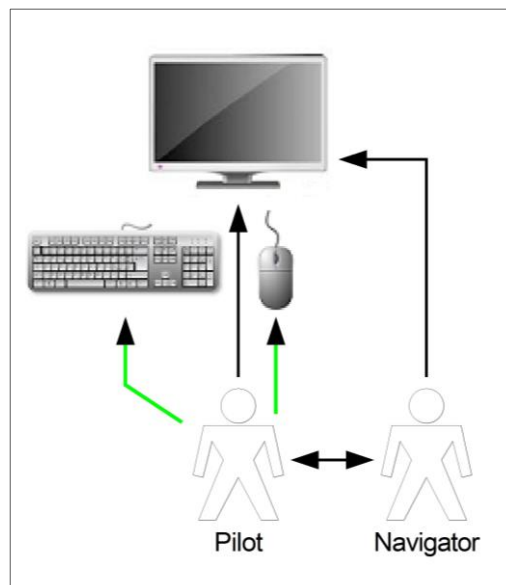
Aus klein wird groß, aber es funktioniert immer!



Änderungen Backlog



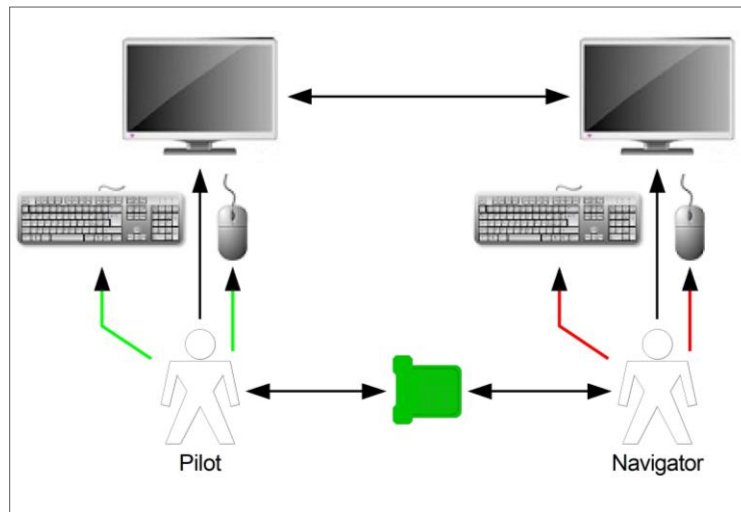
Pair Programming



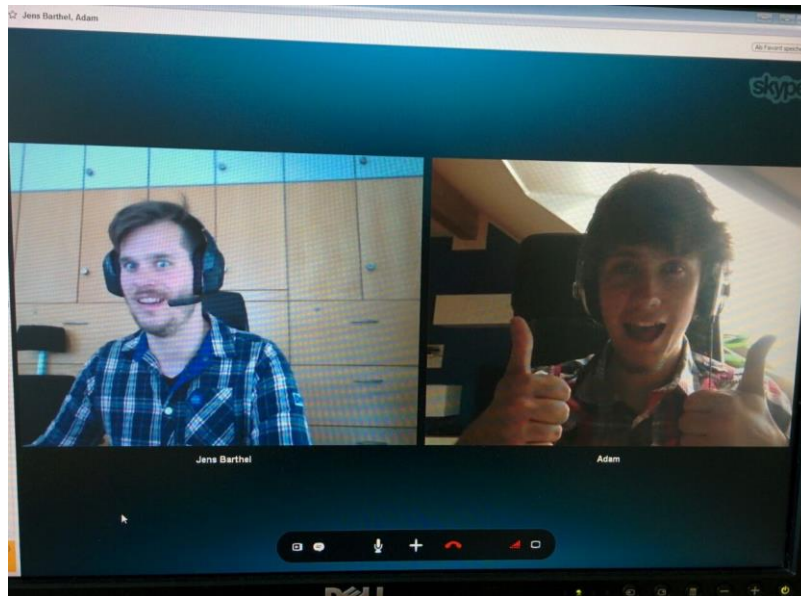
Pair-Programmierung



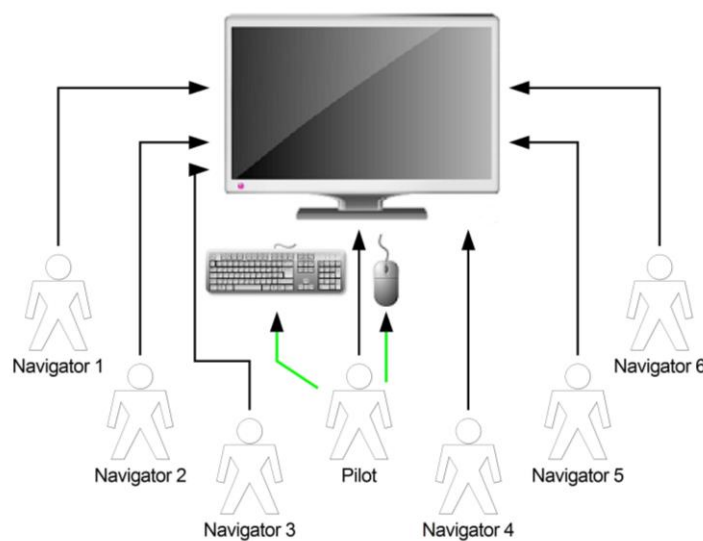
Verteiltes Pair Programmierung



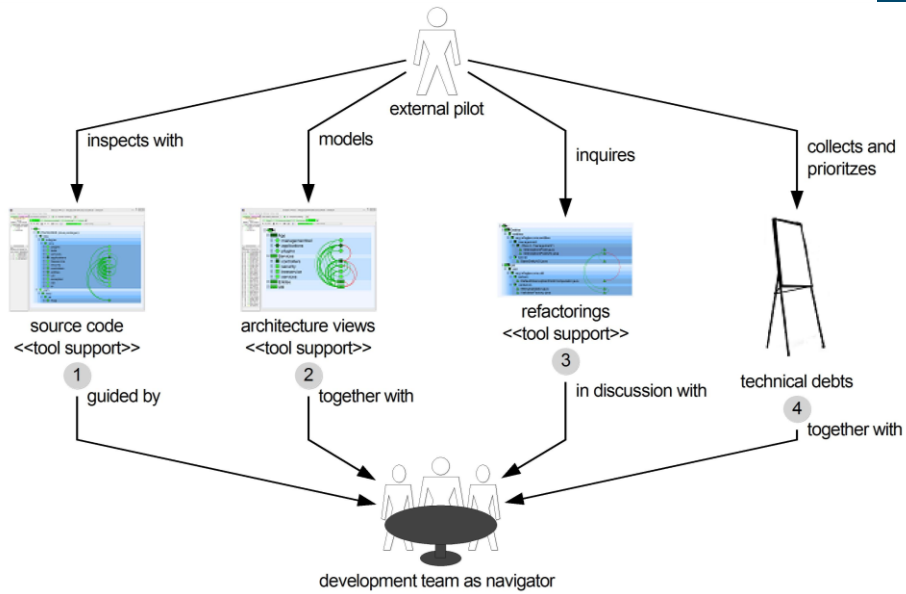
Verteiltes Pair Programming



Mob Programming



Mob Architecting – Architekturanalyse im Team

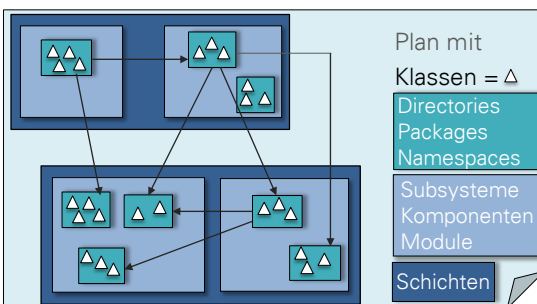


Architekturanalyse: Was ist das?



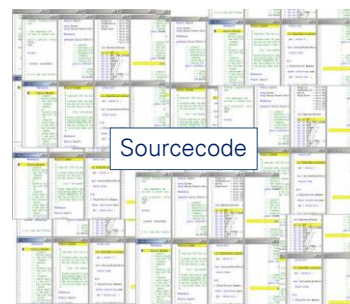
Findet sich die geplante Architektur (Soll-Architektur) in der Strukturen der implementierten Software (Ist-Architektur) wieder?

Soll-Architektur



≠

Ist-Architektur

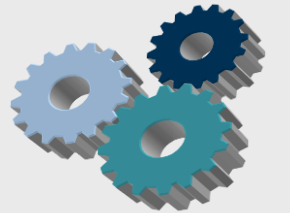


Architekturziele für Langlebigkeit



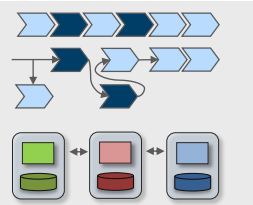
Architekturziel 1: Erweiterbarkeit

- schnelle Anpassungen
- schnelle Fehleranalyse
- Handlungssicherheit
- Stabilität

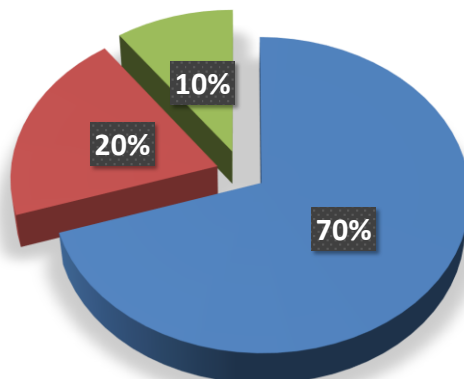


Architekturziel 2: Flexibilität

- Varianten von Geschäftsprozessen
- Geänderte Anforderungen
- Serviceorientierung und Skalierbarkeit
- Baukastenprinzip



Womit verbringen wir unsere Zeit? - revisited



- Code verstehen
- Problem lösen
- Code schreiben

Ausmaß an
technischen Schulden

hohe
Schulden

geringe
Schulden

Wartung und Erweiterung

Architkturerneuerung

Funktionalität
pro Zeiteinheit

Gleichbleibender
Aufwand für Wartung

Ausmaß an
technischen Schulden

hohe
Schulden

geringe
Schulden

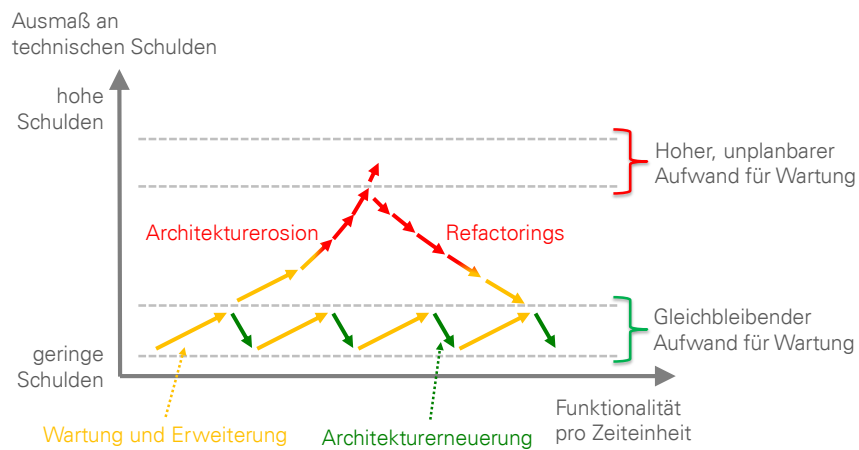
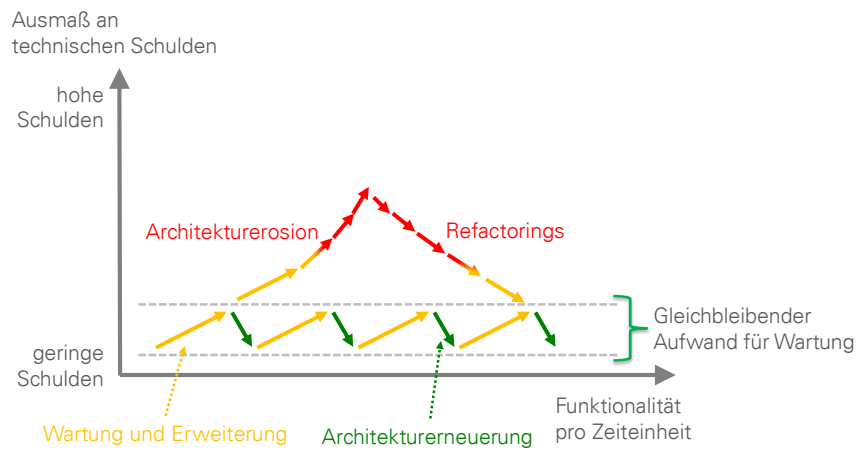
Wartung und Erweiterung

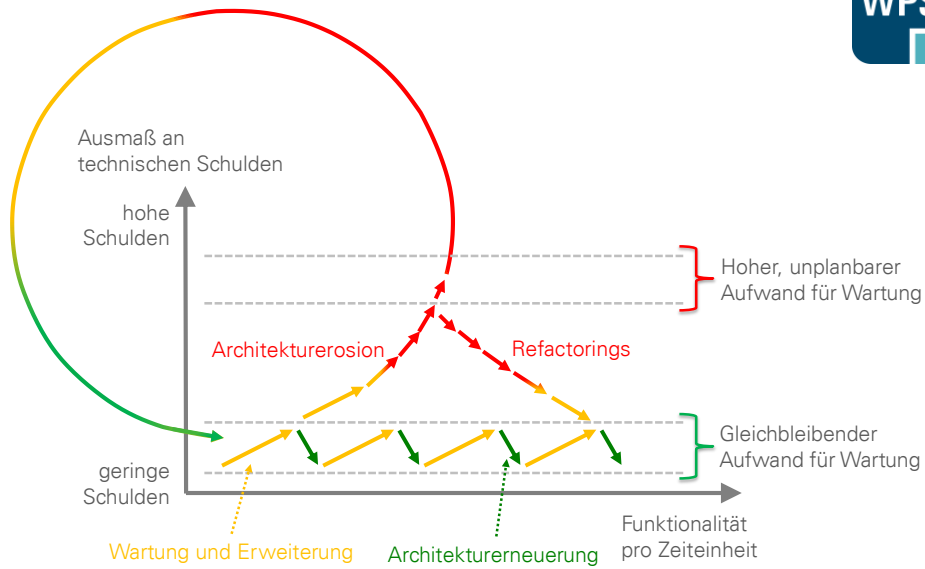
Architkturerneuerung

Funktionalität
pro Zeiteinheit

Gleichbleibender
Aufwand für Wartung

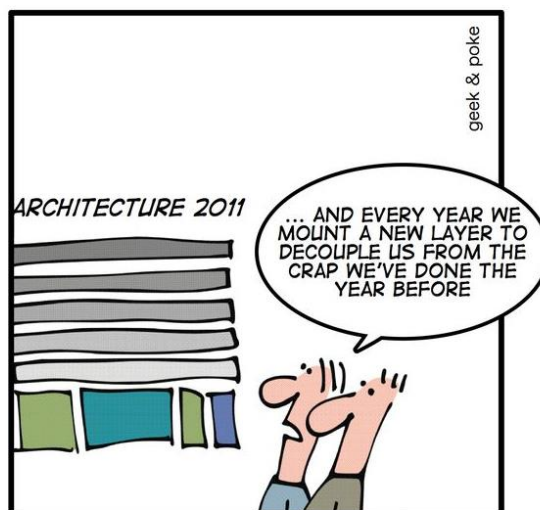
Architekturerosion



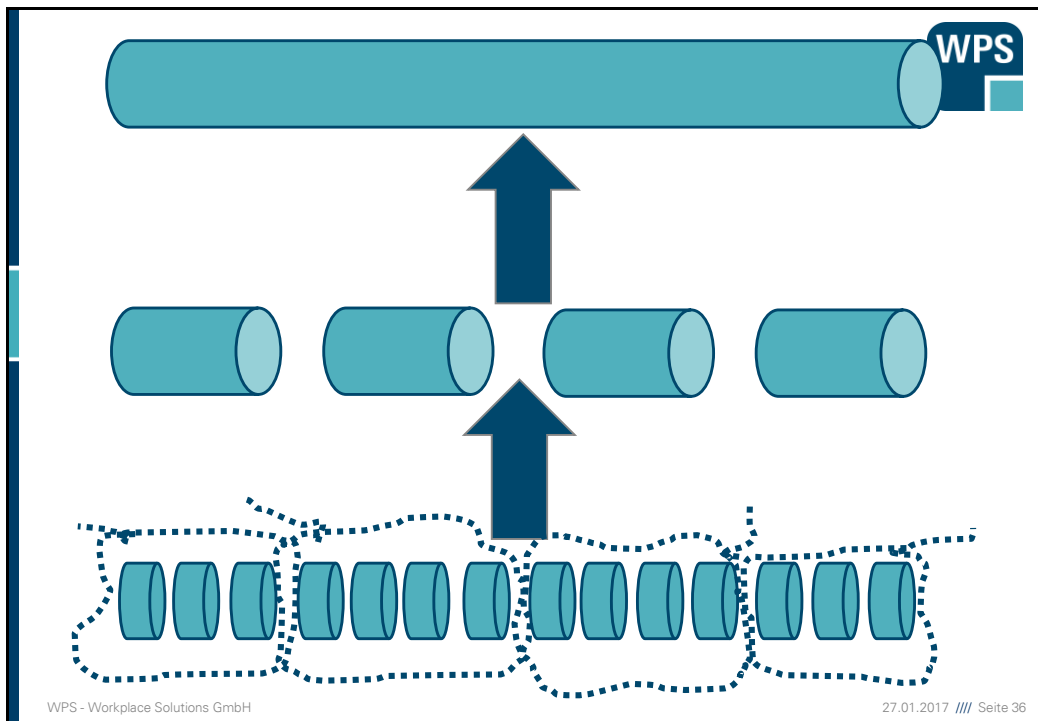
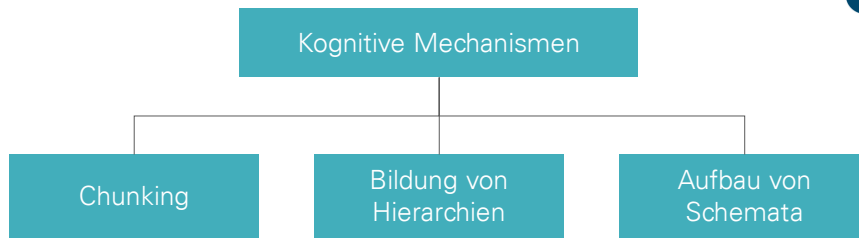


BEST PRACTICES IN APPLICATION ARCHITECTURE

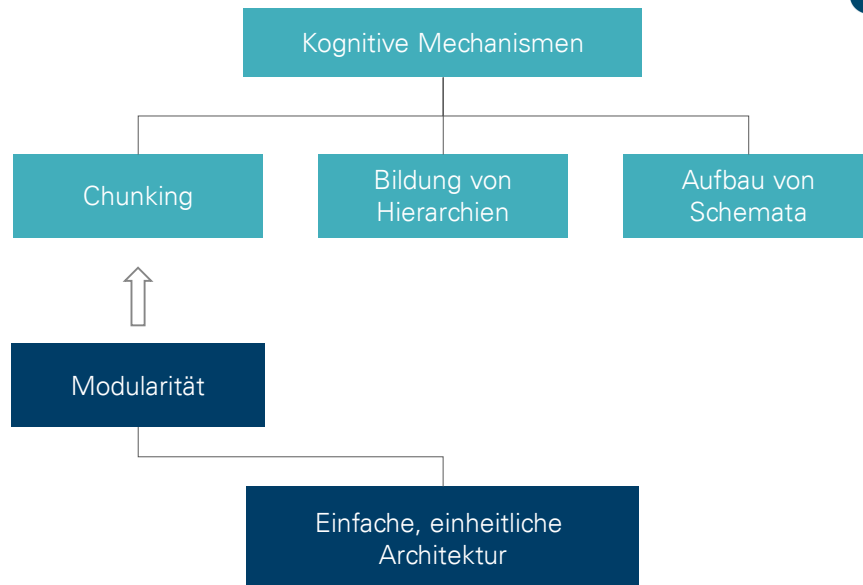
TODAY: USE LAYERS TO DECOUPLE



Komplexe Strukturen verstehen = Zeitgewinn!

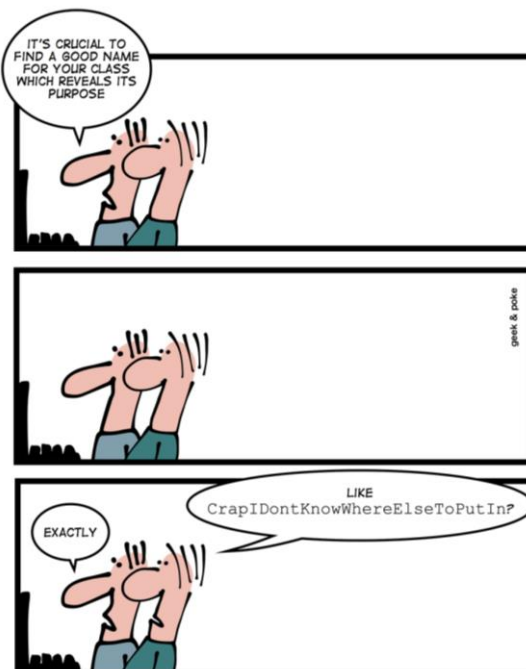
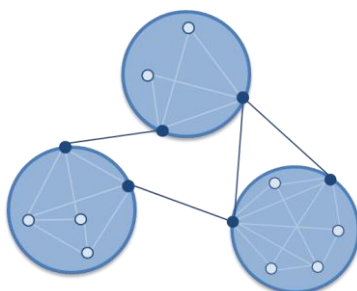


Komplexe Strukturen verstehen = Zeitgewinn!

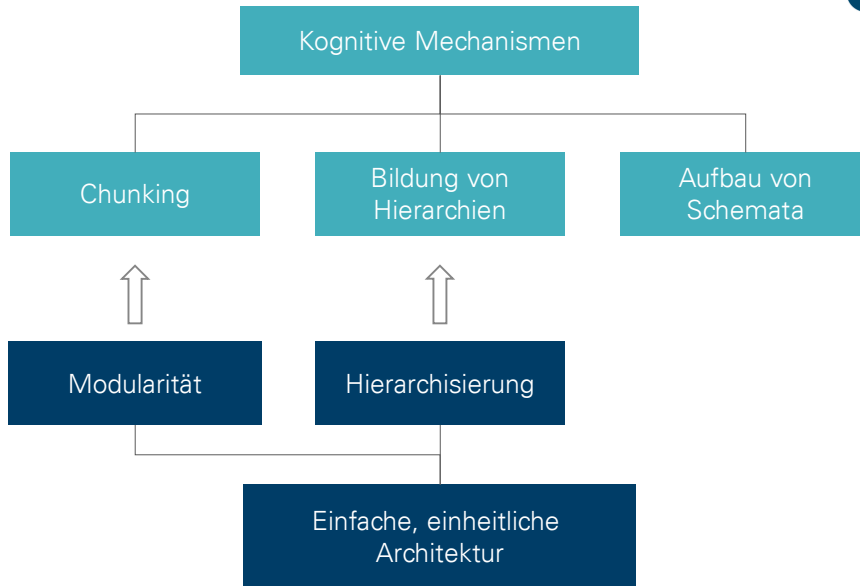


Modularität

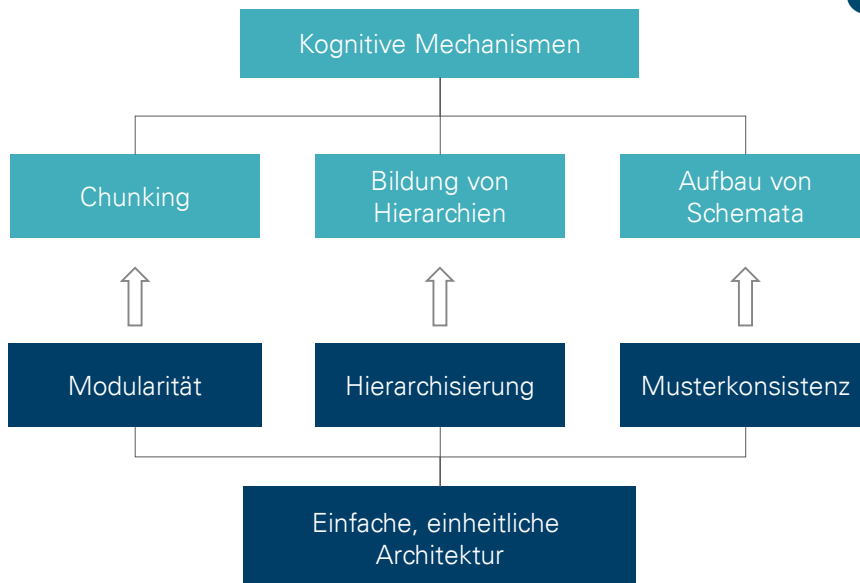
- Hohe Kohäsion und lose Kopplung
- Separation of Concerns
- Single Responsibility Principle



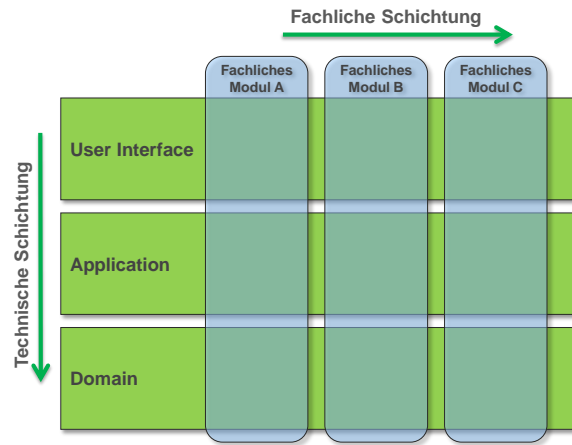
Komplexe Strukturen verstehen = Zeitgewinn!



Komplexe Strukturen verstehen = Zeitgewinn!



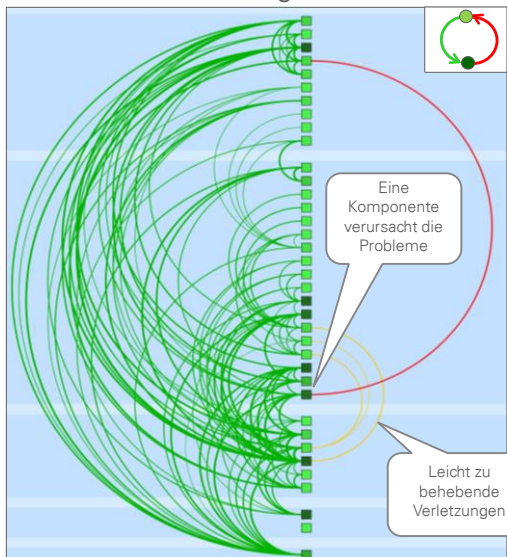
Hierarchien in Architekturebene: Schichten und Module



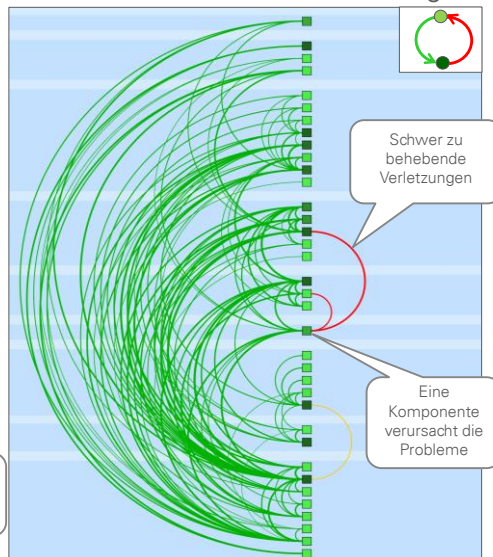
Zwei Dimensionen einer Architektur

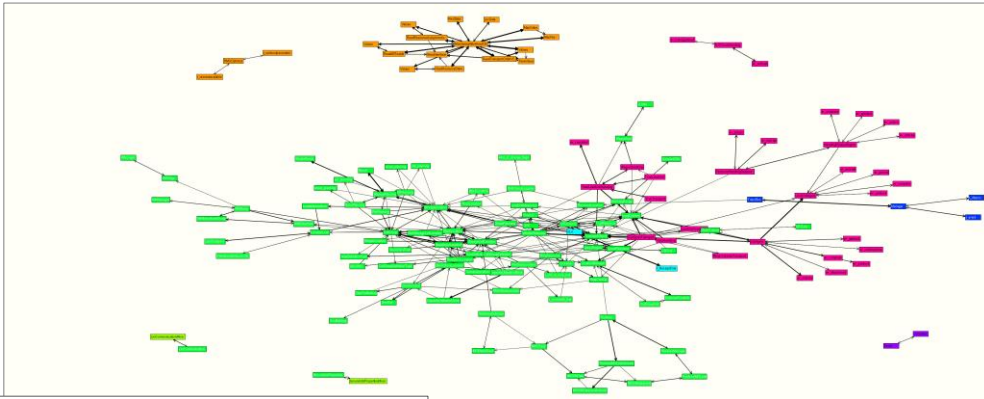


Technische Schichtung



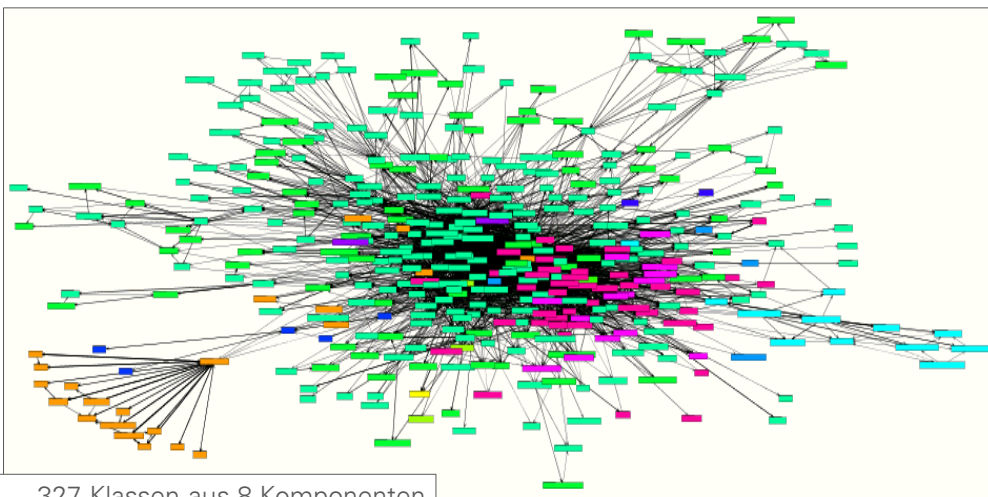
Fachliche Schichtung





119 Klassen aus 4 Komponenten
+ 28 weitere Klassen

Große Zyklen sichtbar machen



327 Klassen aus 8 Komponenten
brauchen sich gegenseitig

WPS

I CANNOT REPRODUCE THE BUG FROM YESTERDAY

MAYBE A TIME-SPACE-DISCONTINUITY HAS COME BY AND MOVED US INTO A PARALLEL UNIVERSE WHERE THIS BUG HAS NEVER EXISTED

geek & poke

SOUNDS REASONABLE. I CLOSE THIS TICKET

WPS - Workplace Solutions GmbH

27.01.2017 /// Seite 61

Komplexe Strukturen verstehen = Zeitgewinn!

WPS

Kognitive Mechanismen

Chunking

Bildung von Hierarchien

Aufbau von Schemata

Modularität

Hierarchisierung

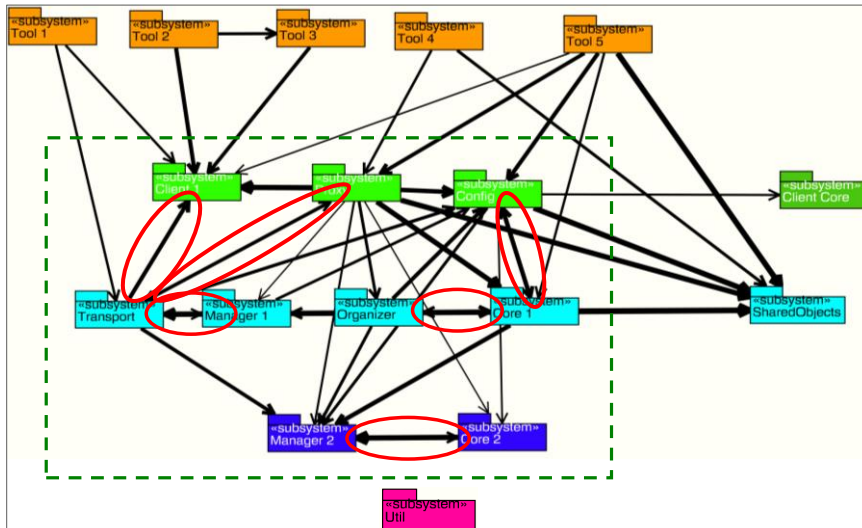
Musterkonsistenz

Einfache, einheitliche Architektur

WPS - Workplace Solutions GmbH

27.01.2017 /// Seite 62

Uneven Modules

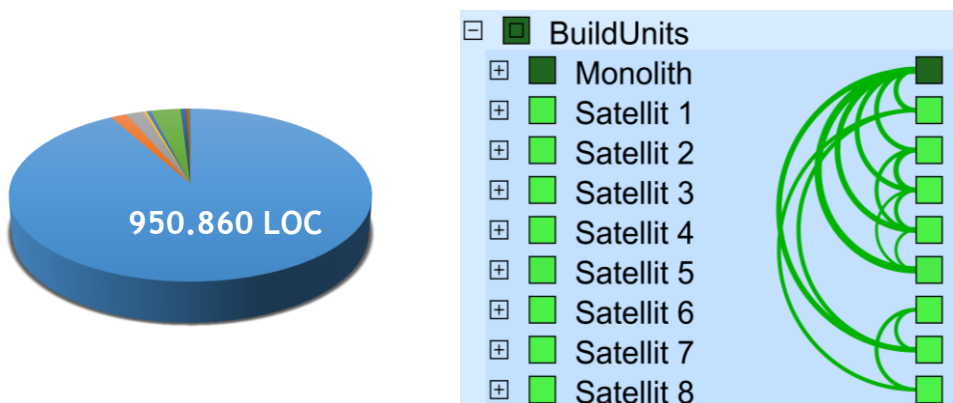


9 Komponenten = 17 Subsysteme

Uneven Modules



Ein großer Monolith mit vielen kleinen Satelliten

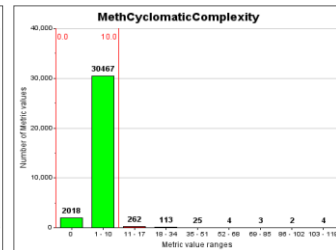
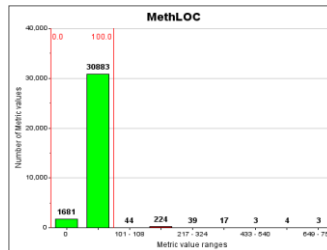
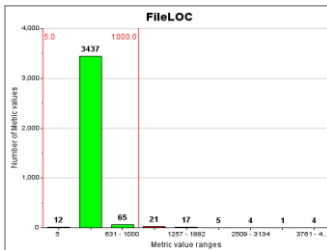


Häufige Ursache: Build-System, das Zyklen verbietet

Modularität: Ausgewogene Größenverhältnisse



- Ist das System auf den verschiedenen Ebenen ausgewogen?
- Welche Code-Abschnitte fallen durch ihre Größe besonders auf?

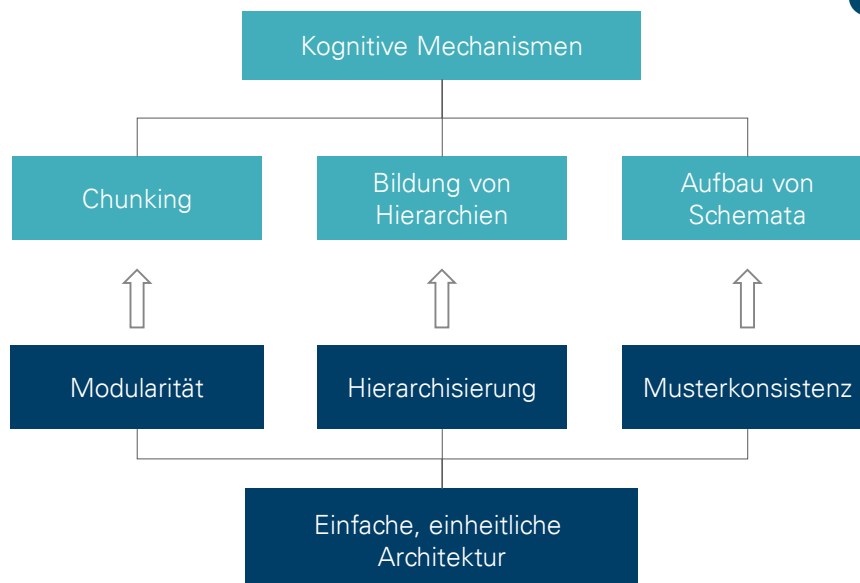


Typische Metriken:

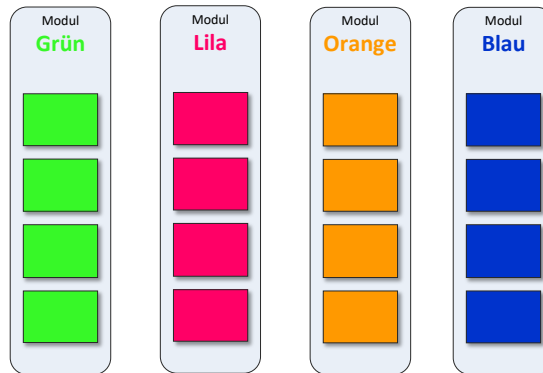
- LOC pro Methode, Klasse, Package, Komponenten
- Duplizierter Code
- Zyklomatische Komplexität



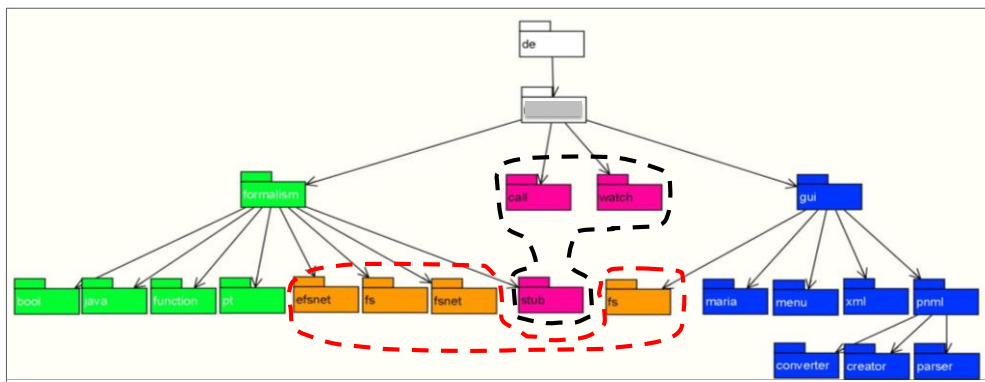
Komplexe Strukturen verstehen = Zeitgewinn!



Muster auf Architekturebene: Vier Module

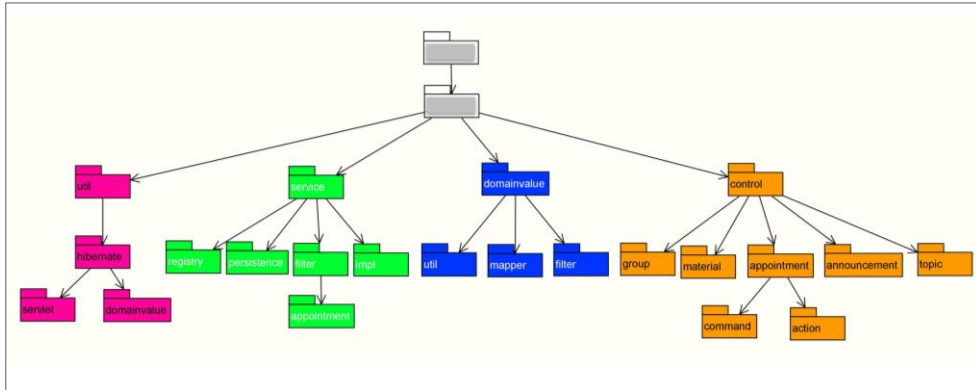


Musterkonsistenz: Was finden wir?

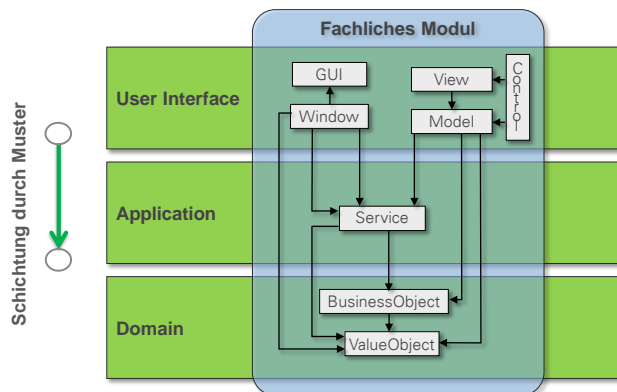


- Ist die Abbildung der Architektur in der Struktur des Codes zu erkennen?

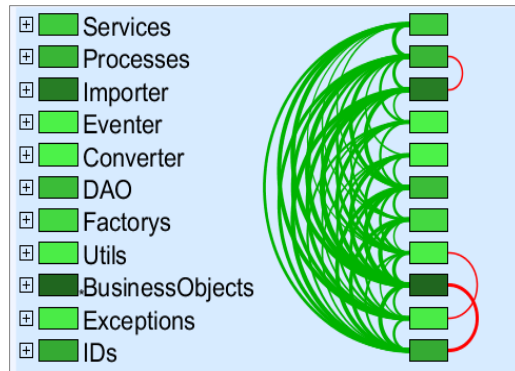
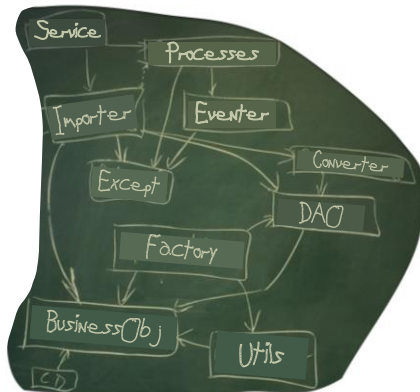
Muster sinnvoll eingesetzt



Muster auf Klassenebene: Entwurfsmuster und Mustersprachen



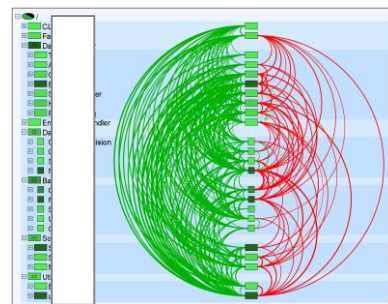
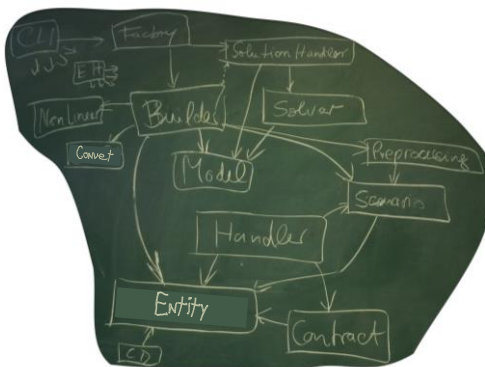
Gute umgesetzte Mustersprache



😊 90% des Sourcecodes lässt sich den Mustern zuordnen

😊 0,1% Verletzungen in den Mustern

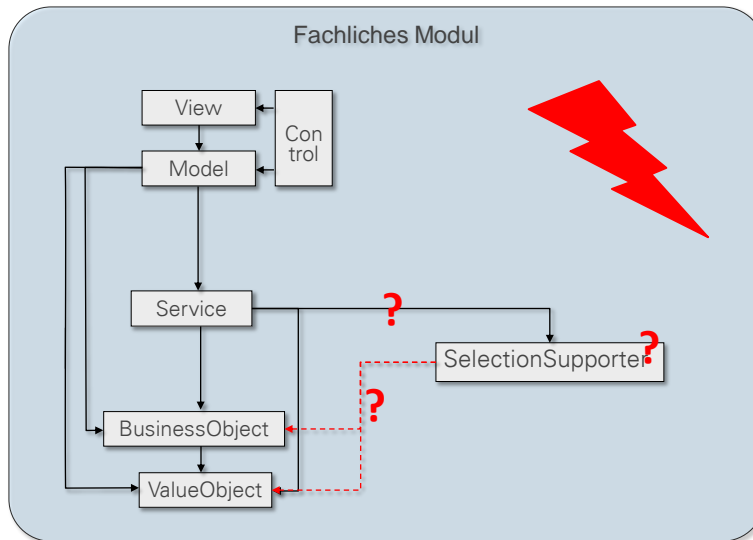
Entdeckung einer Mustersprache



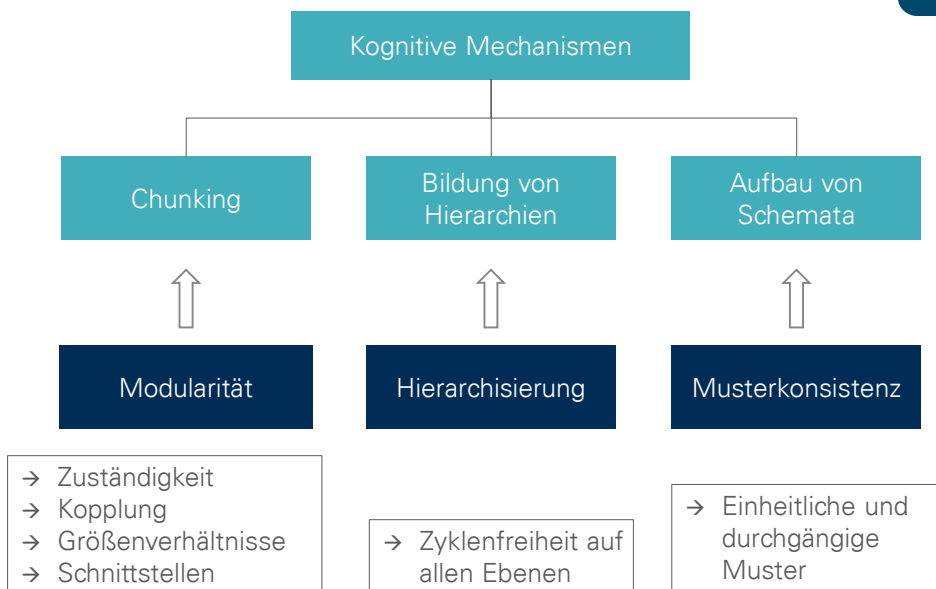
😊 80% des Sourcecodes lässt sich den 23 Mustern zuordnen

😊 4% Verletzungen in den Mustern

Fehlende und unbekannte Muster



Komplexe Strukturen verstehen = Zeitgewinn!



Kostenfreie Werkzeuge

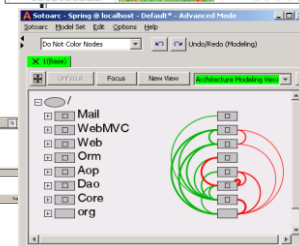
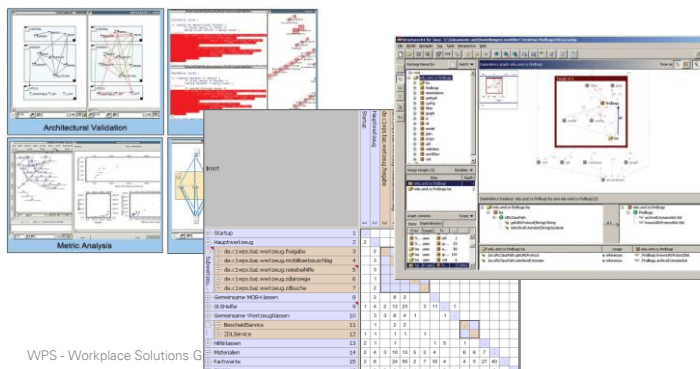


- SonarQube:
 - Leitstand für Qualitätsmetriken
 - Plattform für vielfältige Plugins
- JDepend:
 - wenige Metriken
 - einfache Abhängigkeitsanalyse
- JDepend + Google Architecture Rules:
 - einfache Architekturbeschreibung
- Ndepend/CDepend:
 - Metriken
 - Abhängigkeitsanalyse
- XRadat:
 - Analyse von Java-Projekten via maven
 - Reports bezüglich Komplexität und Architekturverletzungen

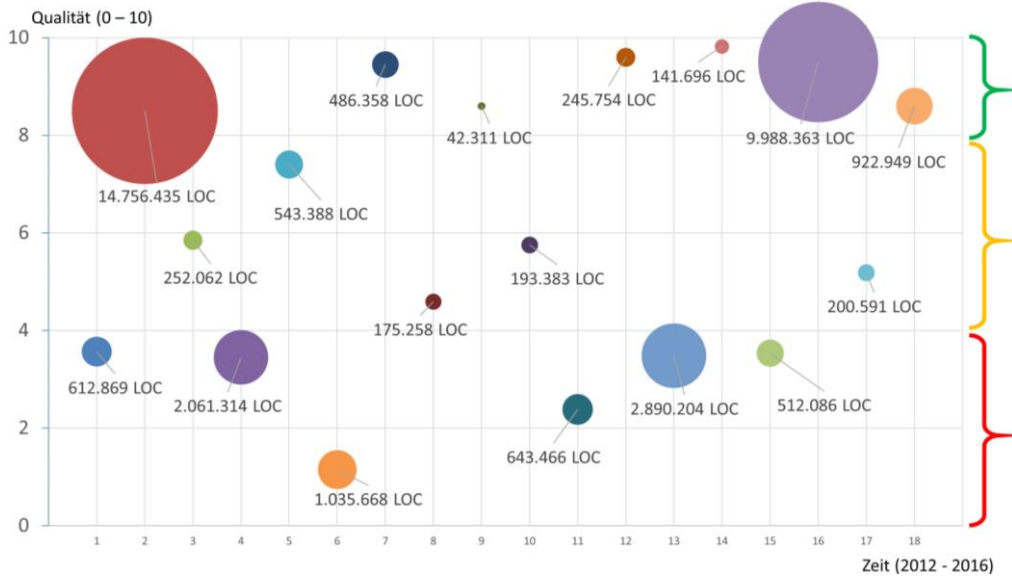


Kommerzielle Produkte

- Axivion Bauhaus: Java, C#, C/C++, Ada, VB und Cobol
- Lattix: Java, .Net, C/C++, Ada, Delphi und DB-Systeme
- Structure101: Java, C++, Ada
- SotoArc und Sonargraph: Java, C#, C/C++, ABAP, PHP
- Teamscale: C#, Java
- Seerene: Java, C#, C/C++, ABAP



Qualität von Softwarearchitekturen - Java, C#, C++, ABAP, PHP -



WPS - Workplace Solutions GmbH

27.01.2017 /// Seite 82



System-Informationen
Dr. Carola Lilienthal
Geschäftsführerin

WPS - WORKPLACE SOLUTIONS

WPS - Workplace Solutions GmbH
Hans-Henny-Jahn-Weg 29
22785 Hamburg

Telefon +49 40 239 489-0
Telefax +49 40 239 489-299
Mobil +49 170 184 77 11

cl@wps.de
www.wps.de

@caiolali

@caiolali
cl@wps.de
www.lisa.de

WPS - Workplace Solutions GmbH

27.01.2017 /// Seite 84

Schulungen zu Architektur und Domain-Driven Design



wps.de/schulung