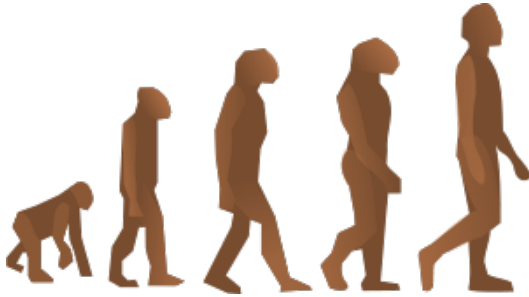


Darwin

Warten auf Godot



(Samuel Beckett, 1952)

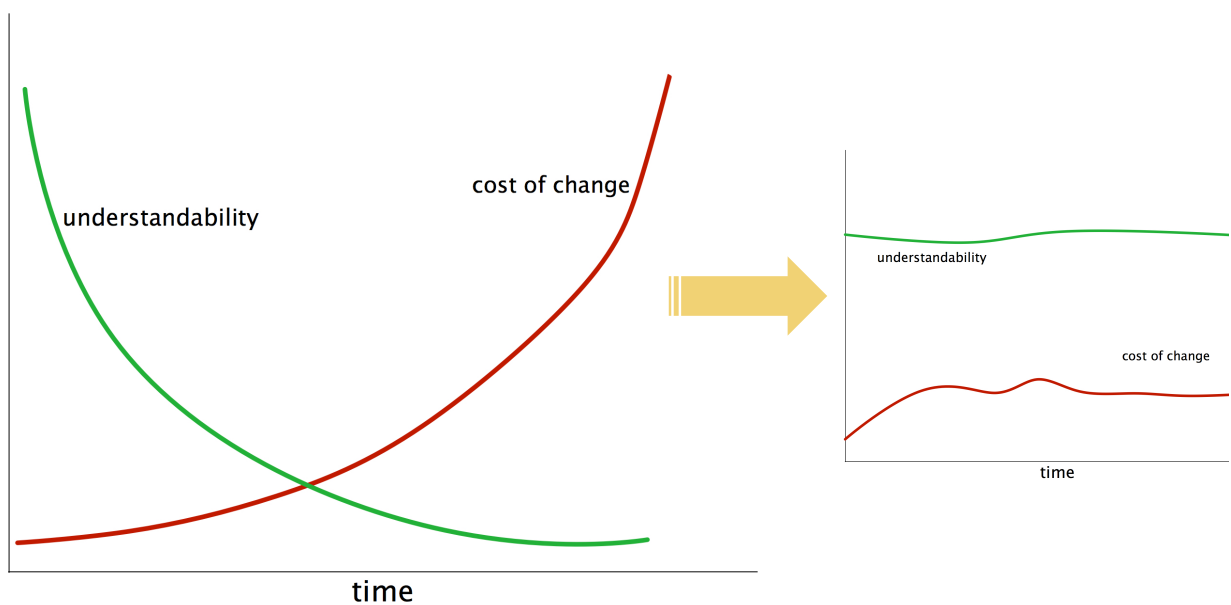
Kategorie „Absurdes Theater“

Gernot Starke

 Follow @gernotstarke

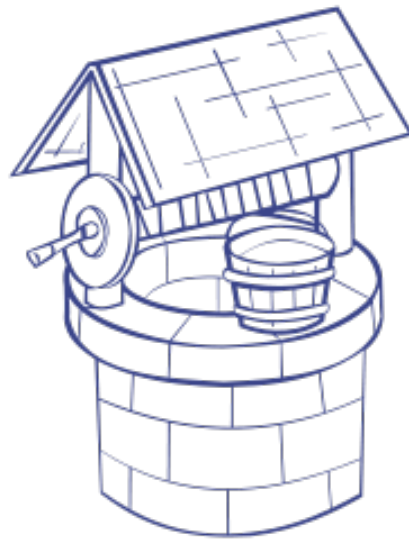
© 2014 Dr. Gernot Starke / innoQ / aim42

Das Problem...



© 2014 Dr. Gernot Starke / innoQ / aim42

Informatiker müssen angeln...



Kinder aus Brunnen angeln
(„retten“) ...

These:
Informatik
Ausbildung fokussiert
von Systemen
auf Neubau

80 : 20 Regel

- ▶ 80% unserer Zeit ändern wir, 20% bauen wir neu.

In der Ausbildung:

- ▶ 100% der Zeit lernen wir neu bauen.
- ▶ In der restlichen Zeit lernen wir ändern.

Wartbare Software benötigt „Ordnung“

- ▶ konzeptionelle Integrität
 - ▶ Substantielles Investment in „innere Ordnung“
- ▶ Verständlichen Code
- ▶ Überblicksdokumentation

Informatik

These:

**Praxis benötigt mehr
Änderungskompetenz**^{an Systemen}

These:

**Änderungen an
Systemen sind durch
Geld motiviert**

Gründe für Änderung an Software

- ▶ Neue / geänderte Anforderungen
- ▶ Änderungen im Kontext
 - ▶ Externe Schnittstellen, Datenformate
 - ▶ Technologie
 - ▶ Organisation
- ▶ Aufgetretene Probleme
 - ▶ Fehler
 - ▶ Verletzung von Qualitätsanforderungen
- ▶ Hohe Betriebs- oder Änderungskosten

Geld!

- ▶ Intrinsische Motivation von Entwicklern

These:

**Budgetverantwortliche
ignorieren
Architekturprinzipien**

These:
Verbesserung
ist mehr als Refactoring

an Systemen
einzelner Klassen

© 2014 Dr. Gernot Starke / innoQ / aim42



Architecture Improvement Method

Gernot Starke & aim42-Contributors

<http://aim42.org>

© 2014 Dr. Gernot Starke / innoQ / aim42

Darum aim42

Methodischer Rahmen für Optimierungs- und Veränderungsprojekte

© 2014 Dr. Gernot Starke / innoQ / aim42

Darum aim42

Gibt Sicherheit bei Änderungen

© 2014 Dr. Gernot Starke / innoQ / aim42

Darum aim42

Adressiert Business und Technik

© 2014 Dr. Gernot Starke / innoQ / aim42

Darum aim42

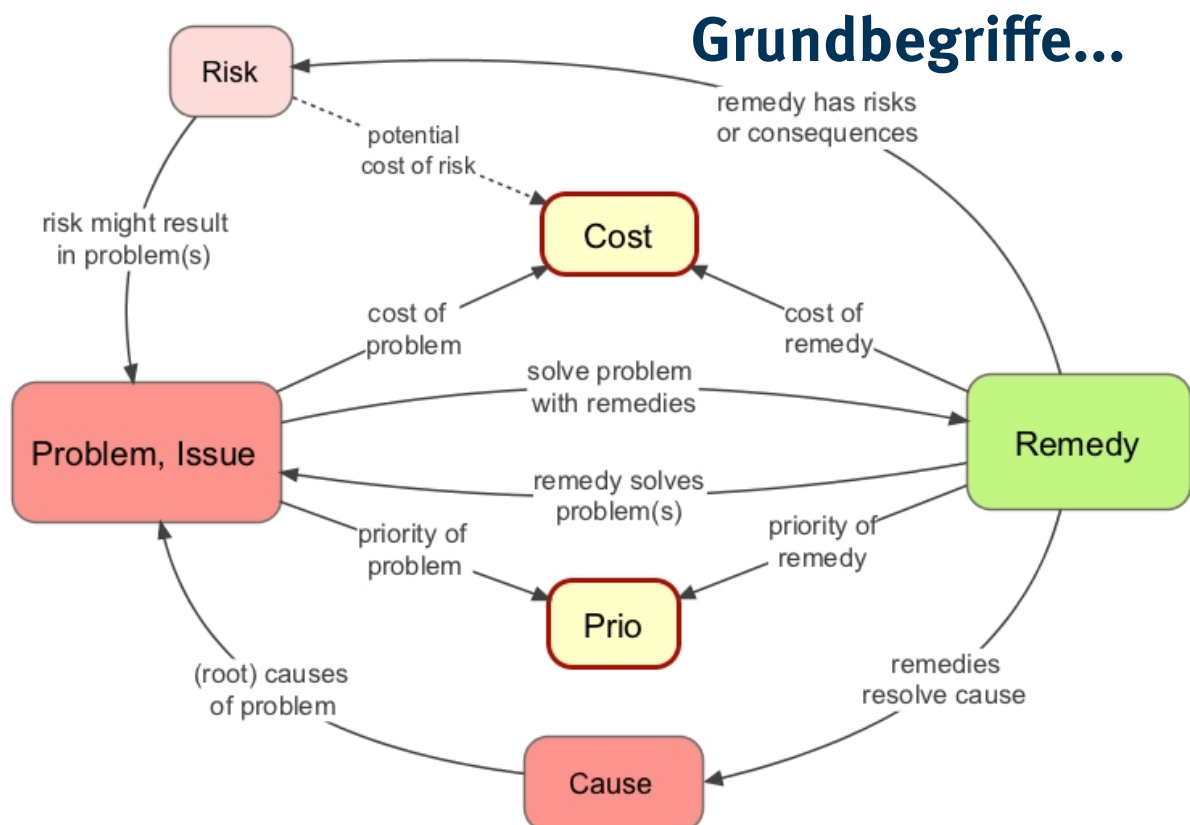
frei, flexibel, open-source

© 2014 Dr. Gernot Starke / innoQ / aim42

Methodik



© 2014 Dr. Gernot Starke / innoQ / aim42



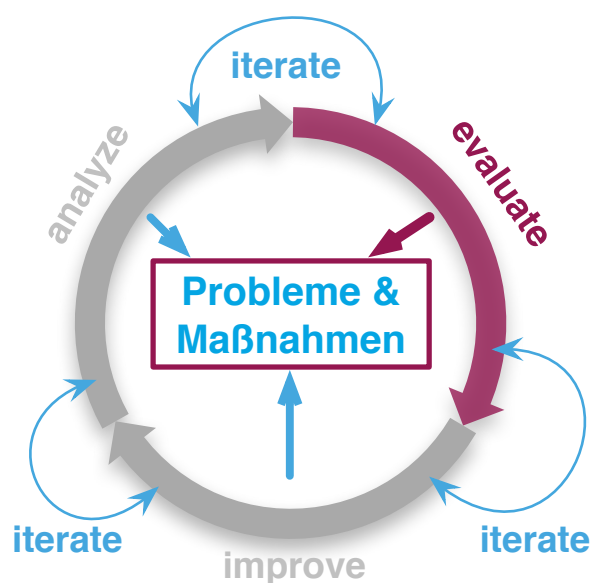
© 2014 Dr. Gernot Starke / innoQ / aim42

Iterative Phased Improvement

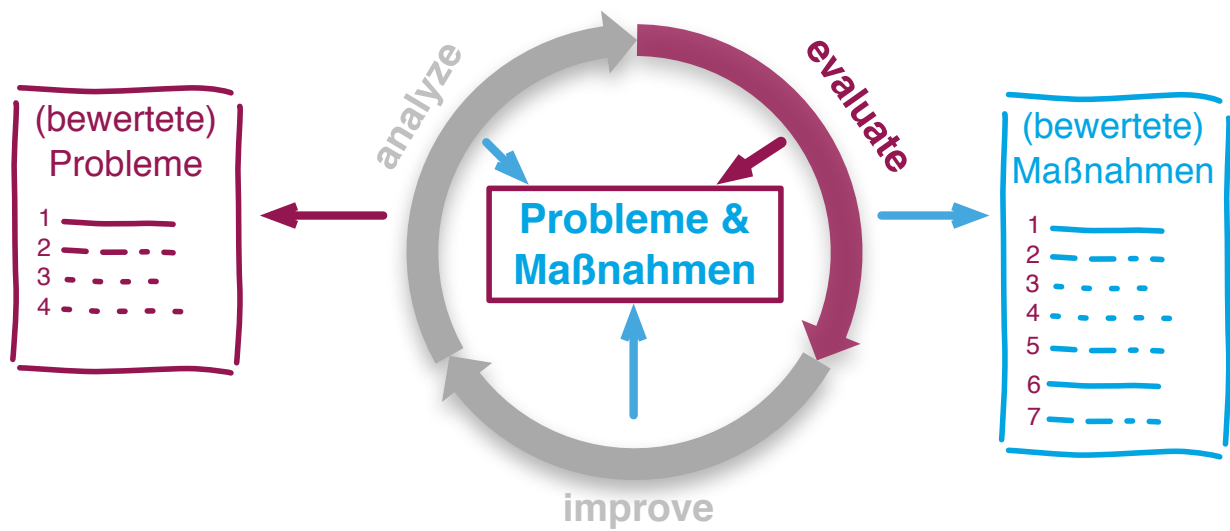
- architecture
- code
- runtime
- organization



Crosscutting-Activities (1)



Crosscutting-Activities (2)



Practices and Patterns



Analyze Practices and Patterns

- ▶ ATAM
- ▶ Capture Quality Requirements
- ▶ Context-Analysis
- ▶ Data-Analysis
- ▶ Development-Process-Analysis
- ▶ Documentation-Analysis
- ▶ Issue-Tracker-Analysis
- ▶ Pre-Interview Questionnaire
- ▶ Profiling
- ▶ Qualitative Analysis
- ▶ Quantitative-Analysis
- ▶ Questionnaire
- ▶ Root Cause Analysis
- ▶ Runtime-Artifact-Analysis
- ▶ Software Archeology
- ▶ Stakeholder-Analysis
- ▶ Stakeholder-Interview
- ▶ Static Code Analysis
- ▶ Use-Case-Cluster
- ▶ View Based Understanding

Analyze Practices and Patterns...

- ▶ ATAM
- ▶ Context-Analysis
- ▶ Development-Process-Analysis
- ▶ Issue-Tracker-Analysis
- ▶ Questionnaire
- ▶ Software Archeology
- ▶ Stakeholder-Interview
- ▶ Static Code Analysis

Sample Practices from ANALYZE

- ▶ **ATAM: Architecture Tradeoff Analysis Method.** Systematic approach to find architectural risks and tradeoffs (compromises) .
- ▶ **DATA ANALYSIS:** Analyse and inspect the data created and manipulated by the system for its content, structure, quantity and size.
- ▶ **PRE-INTERVIEW-QUESTIONNAIRE:** Prior to interviewing stakeholders, present them with a written questionnaire, so they can reflect in advance.
- ▶ **STATIC CODE ANALYSIS:** Analyse source code to identify building blocks and their dependencies, determine complexity, coupling, cohesion and other structural properties.

© 2014 Dr. Gernot Starke / innoQ / aim42

Evaluate Practices and Patterns

- ▶ Estimate in Intervall
- ▶ Estimate Problem Cost
- ▶ Estimate Remedy Cost
- ▶ Failure Mode and Effect Analysis
- ▶ Impact Analysis

© 2014 Dr. Gernot Starke / innoQ / aim42

Estimate-in-Intervalls

- ▶ Schätze IMMER in Intervallen
- ▶ Breite des Intervalls ==
Eigenes Vertrauen in Schätzung

[15..2]

© 2014 Dr. Gernot Starke / innoQ / aim42

Improvement Practices and Patterns

- ▶ Anticorruption-Layer
- ▶ Assertions
- ▶ Automated-Tests
- ▶ Branch-For-Improvement
- ▶ Extract-Reusable-Component
- ▶ Group-Improvement-Actions
- ▶ Improve-Code-Layout
- ▶ Introduce Boy Scout Rule
- ▶ Interface Segregation Principle
- ▶ Isolate Changes
- ▶ Keep-Data-Toss-Code
- ▶ Never-Change-Running-System
- ▶ Quality-Driven-Software-Architecture
- ▶ Refactoring
- ▶ Refactoring-Plan
- ▶ Remove-Nested-Control-Structures
- ▶ Sample-For-Improvement
- ▶ Schedule-Work
- ▶ Untangle-Code
- ▶ Use Invariants To Kill Zombies

© 2014 Dr. Gernot Starke / innoQ / aim42

Improvement Practices and Patterns

- ▶ Automated-Tests
- ▶ Quality-Driven-Software-Architecture
- ▶ Sample-For-Improvement
- ▶ Introduce Boy Scout Rule
- ▶ Isolate Changes

© 2014 Dr. Gernot Starke / innoQ / aim42

Introduce Boy-Scout Rule

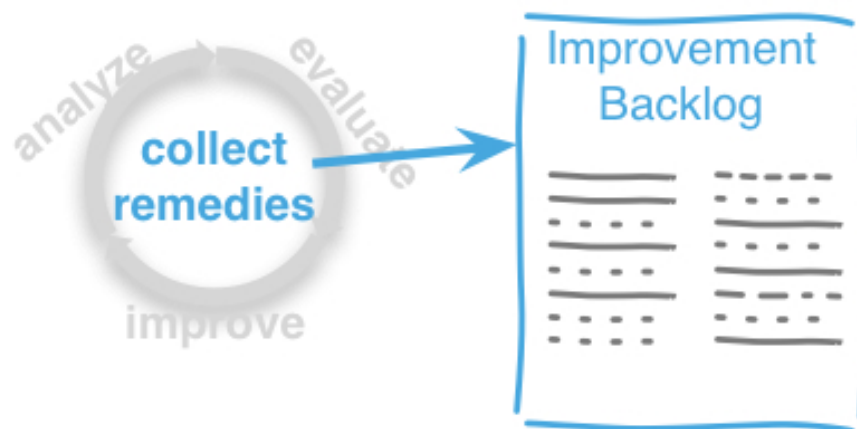
- ▶ Im bestehenden Code die BSC einführen...
- ▶ Klärung:
 - ▶ welche Code-Regeln zukünftig gelten sollen
 - ▶ Prioritäten der Veränderung (z.B. Bezeichner, Vereinfachung, Sonar-Findings, Formatierung)
 - ▶ Kommunikation im / über Code

© 2014 Dr. Gernot Starke / innoQ / aim42

Crosscutting Practices and Patterns

- ▶ Explicit Assumption
- ▶ Fast Feedback
- ▶ Collect Problems
- ▶ Collect Opportunities for Improvement
- ▶ Improvement Backlog

Improvement Backlog



Improvement Backlog (kompakte Fassung)

- ▶ Probleme mit Maßnahmen zur Behebung
- ▶ Inklusive Kosten
 - ▶ Probleme: Kosten pro Zeit/Auftreten
 - ▶ Maßnahmen
- ▶ Risiken der Behebung

Prio	Problem	Maßnahmen (Remedies)	Kosten (Problem)	Kosten (Behebung)	Risiken
1					
2					
3					

Sample Crosscutting Practices and Patterns

- ▶ COLLECT PROBLEMS: Maintain a central list or overview of known problems, together with their cost/effort evaluation.
- ▶ COLLECT OPPORTUNITIES FOR IMPROVEMENT: In all AIM42 phases, one should identify remedies for the currently known problems or their causes.
- ▶ IMPROVEMENT BACKLOG: A list or collection of remedies and their cost/effort/risk estimation.
- ▶ In the sense of lean and agile, teams shall try to FAST
FEEDBACK: The later a lack of quality is identified the higher are the costs to fix it.

Management überzeugen



© 2014 Dr. Gernot Starke / innoQ / aim42

Management überzeugen

You need to talk business!

(Martin Fowler, OOP 2014)



© 2014 Dr. Gernot Starke / innoQ / aim42

Management überzeugen...

- ▶ Problem Cost ermitteln
 - ▶ Technische Probleme haben einen Preis
 - ▶ Schätzen mit expliziten Annahmen

© 2014 Dr. Gernot Starke / innoQ / aim42

Beispiel: Mehraufwand „Heterogenität“

- ▶ Technologiezoo: System aus >20 Subsystemen in 8 Technologien
- ▶ Techniker: „aufwändig, komplex“
- ▶ Management: was kostet das?

© 2014 Dr. Gernot Starke / innoQ / aim42

Kosten der (übertriebenen) Heterogenität

- ▶ Mehraufwände in Lebenszyklus-Phasen schätzen
 - ▶ Analyse, Architektur, Implementierung, Test, Betrieb
 - ▶ Unterstützt durch reale Aufwandsmessungen

Mehrkosten „Heterogenität“ [20%..2%]

	Anteil	Mehraufwand		1.000 € min		
		min	max	min	max	max
				1.017,78 €	1.204,56 €	
Requirements	7%			70 €	70,00 €	70,00 €
Design/Architektur	6%			60 €	60,42 €	61,20 €
10% Zusatzaufwand Schnittstellen		5%	15%		0,30	0,90
10% übergreifende Entscheidungen		2%	5%		0,12	0,30
80% Sonstiges						
Programmierung	12%			120 €	122,40 €	145,68 €
2% Setup, Updates-Umgebung		5%	100%		0,12	2,40
2% Einarbeitung, Recherche		5%	20%		0,12	0,48
10% Fehlersuche, Testing		3%	100%		0,36	12,00
5% Effiziente Lösung von Detailproblem		-10%	-40%		-0,60	-2,40
10% Lösung von Standardproblemen		10%	50%		1,20	6,00
20% Team-interne Abstimmung		5%	30%		1,20	7,20
51% Sonstiges						
Integration / Test	8%			80 €	83,40 €	113,80 €
5% Komponenten integrieren		5%	100%		0,20	4,00
30% Integrationstests durchführen		5%	50%		1,20	12,00
20% Integrationstests auswerten		10%	50%		1,60	8,00
10% Testumgebung bereitstellen/erhalten		5%	80%		0,40	6,40
35% Sonstiges						
Maintenance / Operations	67%			670 €	681,56 €	813,88 €
3% Vorhalten von Entwicklerkapazität		5%	20%		1,01	4,02
5% Entwickler finden/einarbeiten		10%	30%		3,35	10,05
1% Versions- und Security-Updates		3%	10%		0,20	0,67
1% Auswahl / Beschaffung Laufzeitumgebungen		10%	100%		0,67	6,70
3% Konfiguration, Installation		5%	70%		1,01	14,07
0,50% Monitoring, Logging		5%	10%		0,17	0,34
5% Fehlersuche und -Behebung		1%	100%		0,34	33,50
2% Skalierung/Clustering		5%	15%		0,67	2,01
1% Paketierung, Deployment-Vorbereitung		2%	10%		0,13	0,67
30% Erweiterungen/Änderungen vornehmen		2%	30%		4,02	60,30
49% Sonstiges						

Praxis



© 2014 Dr. Gernot Starke / innoQ / aim42

Praktisch eingesetzt...

- ▶ Automotive:
„Multimedia-Framework“
- ▶ Rail-Service
„Infrastruktur“
- ▶ Mobilfunk
„Billing“
- ▶ Airport-Operations
„Luggage Handling“
- ▶ Systemsoftware für
Maschinenbau /
Lebensmittelindustrie
- ▶ 2014:
 - ▶ Europäische Bahn
(Audit OnlineTicket)
 - ▶ ERP-Hersteller
(Audit und Rebuild
Kernsystem)

© 2014 Dr. Gernot Starke / innoQ / aim42

Projekt



© 2014 Dr. Gernot Starke / innoQ / aim42

Build & DevelopmentProcess

- ▶ **Method Guide:** AsciiDoc
- ▶ Gradle, Travis-CI
- ▶ Ergebnis:
 - ▶ HTML: <http://aim42.github.io>

© 2014 Dr. Gernot Starke / innoQ / aim42

Website

software architecture improvement

HOME APPLY READ FAQ CONTRI

 software evolution and optimization - done right

aim42 - systematic software evolution and improvement

- Optimize your software and reduce maintenance cost
- Control risks, issues and technical debt
- Organizes patterns and practices in three phases
- Free and open-source
- Grounded in practice, proven approaches, backed by serious research

See the online aim42 method guide for details

evaluate

estimate „value“ of problems, issues and their remedies, prioritize.

more ...

improve

systematically improve code and reduce technical debt, remove ...

more ...

Download Whitepaper



news

- Whitepaper available
- April 2014: aim42 experience part of the industry track of the Conference on Software Arc
- Feb. 2014: Online version of method guide
- Feb. 2014: Presentation on a OOP conference (Stefan Tilk)

Whitepaper



Architecture Improvement Method

Methodical Improvement of Software Systems and -Architectures

Dr. Gernot Starke
<http://aim42.org>

This paper outlines aim⁴², the architecture improvement method, a systematic yet pragmatic approach to improve productive software systems and architectures. aim⁴² relies on a small number of domain concepts and works iteratively in three phases (analyze, evaluate, improve) supported by crosscutting activities. For each phase, aim⁴² proposes a number of proven and established practices and patterns. The method addresses both business and technical stakeholders of software systems. aim⁴² is developed by an active community in open-source style, backed by extensive industrial experience and scientific research. It has proven to work under time and budget constraints in various industries.

1 Introduction

Real-world software systems regularly need to be maintained for various reasons, often under severe budget and time constraints. Software owners and other stakeholders often prioritize new business requirements higher than improvement of internal software quality. Over time, this leads to reduced maintainability, coined "software erosion" or "software entropy".

Especially in competitive markets, investment in existing software is often driven by short-term business goals. Long-term goals, like maintainability or understandability are neglected. Improvements of software architectures seem to conflict with these short-term business and budget requirements, as break-even is expected within short timespans.

Keeping software maintainable over time requires substantial investment in internal qualities, like conceptual integrity, architectural structures and crosscutting concepts, proper coupling and cohesion.

The systematic approach of aim⁴² supports evolution and improvement of software architectures and internal quality. aim⁴² helps technical and management decision makers to properly compromise short-term budget requirements with long-term internal architecture quality.

Software Architecture Improvement - Whitepaper

© 2014 Dr. Gernot Starke / innoQ / aim42

Code: github

public repo for aim42, especially the "aim42 method guide" — Edit

170 commits 2 branches 0 releases 6 contributors

Your recently pushed branches:

atam (6 minutes ago)

Compare & pull request

branch: master

aim42

Merge branch 'atam'

gernotstarke authored 11 hours ago latest commit b2383582b5

graphics

added image for "view-based-understanding"

6 days ago

guide

fixing list

21 hours ago

whitepaper

improved layout of whitepaper

3 days ago

.gitignore

Merge branch 'atam'

11 hours ago

.travis.yml

Added encrypted key for repository aim42/aim42

2 months ago

readme.md

Improved image links in Readme.

2 months ago

readme.md



Architecture Improvement Method

© 2014 Dr. Gernot Starke / innoQ / aim42

Issues: viele...

Assigned to you9

Created by you15

Mentioning you1

No milestone selected

Labels

Fundamental4

enhancement6

pattern10

practice10

question3

website1

bug0

duplicate0

wontfix0

Manage labels

New label

New label name

CloseLabelAssigneeMilestone

Convert OOP-2014 talk on business-relation of architecture into aim42 pattern enhancement pattern#53

Opened by gernotstarke 6 days ago

Write pattern "shepharding the architecture" pattern practice#52

Opened by gernotstarke 6 days ago

Estimate-in-Intervalls (new practice for evaluation phase) practice#51

Opened by gernotstarke 8 days ago 3 comments

Agree on a name (symbol) for the "system under improvement" Fundamental question#50

Opened by MichaelMahlberg 24 days ago 1 comment

expand "code review" or "code reading" practice practice#48

Opened by gernotstarke a month ago

enhance build to generate pdf version enhancement#46

Opened by gernotstarke a month ago

Cross-Check our patterns/practices with "OORP" from Nierstrasz et al enhancement Fundamental#44

Opened by gernotstarke a month ago 1 comment

Include "architectural feature" from SAFE-framework enhancement pattern#43

Opened by gernotstarke a month ago

Add pattern for better logging and runtime metrics under "Improve"#38

Opened by vanto 2 months ago

What is technical debt? (Warning: Meta Question) Fundamental question#36

© 2014 Dr. Gernot Starke / innoQ / aim42

Graph

Members

Contributors...

Members of the aim42 Network

aim42 created aim42 and everyone else forked it. This is the family tree.

aim42 / aim42

aheusingfeld / aim42

BoronNitride / aim42

ck-innoq / aim42

feststelltaste / aim42

gbeine / aim42

httpPrincess / aim42

MichaelMahlberg / aim42

otigges / aim42

rschimmack / aim42

vanto / aim42

Beiträge /
pull-requests
Willkommen!

© 2014 Dr. Gernot Starke / innoQ / aim42

Publicity



© 2014 Dr. Gernot Starke / innoQ / aim42

Publicity... (1): BT-Magazin Mai 2014

Business Technology

Architektur, Innovation & Strategie

Business Technology, das Magazin für IT-Architektur und -Management für Architekten, IT-Manager und Berater, die sich auf dem Vorstandsparkett ebenso sicher bewegen wie auf dem Marktplatz der Technologien und Architekturkonzepte. Business Technology bietet klar und verständlich aufbereitete Architekturthemen mit einem übergreifenden Blickwinkel. Das Magazin erläutert praxisnah und kompetent, welche konzeptionellen und technologischen Trends die Unternehmenswelt heute und in Zukunft verändern und legt dabei größten Wert auf qualitativ hochwertige Informationen.. Business Technology positioniert sich durch praxisnahe Expertenbeiträge, gut recherchierte Hintergrundberichte, spannende Case Studies sowie Interviews mit Top-Führungskräften.



Aktuelle Ausgabe



Archiv



Digital



ABO Abonnement

© 2014 Dr. Gernot Starke / innoQ / aim42

Publicity... (2): ECSA 2014 (August)

European Conference on Software Architecture



Banner © by ECSA Conference

ECSA 2014

The European Conference on Software Architecture (ECSA) is the premier European software architecture conference, providing researchers, practitioners, and educators with a platform to present and discuss the most recent, innovative and significant findings and experiences in the field of software architecture research and practice. In 2014, the conference will feature keynotes, research track, industrial track, workshops, tutorials, doctoral symposium, and tool demonstrations and poster presentations.

ECSA 2014 will take place at the University of Vienna, Austria, from August 25 to 29, 2014.

© 2014 Dr. Gernot Starke / innoQ / aim42

Contributions welcome

- ▶ Method guide: **<http://aim42.github.io>**
 - ▶ Source: <https://github.com/aim42/aim42>
 - ▶ <https://github.com/aim42/aim42/issues>
- ▶ Twitter: @arc_improve42
- ▶ Mailing list: aim42@lists.innoq.com

© 2014 Dr. Gernot Starke / innoQ / aim42

Disclaimer & Legal Notice

- ▶ Graphics by
 - ▶ openclipart.org
 - ▶ aim42.org
- ▶ Licensed under Creative Commons Sharealike 4.0
- ▶ <https://creativecommons.org/licenses/by-sa/4.0/>
- ▶ aim42 logo by Gernot Starke