



Orientation in Objects

Kontinuierliche Architekturdokumentation im agilen Umfeld

Orientation in Objects GmbH

Weinheimer Str. 68
68309 Mannheim

www.oio.de
info@oio.de



Falk Sippach (@sipp sack)

Trainer, Berater, Entwickler



Co-Organisator

*Architektur
Agile Softwareentwicklung
Codequalität*



Feedback gern an: @sipp sack, falk.sippach@oio.de

Java und XML

) Software Factory)

- Schlüsselfertige Realisierung von Java Software
- Individualsoftware
- Pilot- und Migrationsprojekte
- Sanierung von Software
- Software Wartung

) Object Rangers)

- Unterstützung laufender Java Projekte
- Perfect Match
- Rent-a-team
- Coaching on the project
- Inhouse Outsourcing

) Competence Center)

- Schulungen, Coaching, Weiterbildungsberatung, Train & Solve-Programme
- Methoden, Standards und Tools für die Entwicklung von offenen, unternehmensweiten Systemen



Man kann zwar an vielen Stellen nachlesen, wie man Architekturdokumentation strukturiert. Aber auf der Suche nach einer praktikablen Handhabung zur Erstellung und Pflege enden die meisten Versuche in der WYSIWYG-Hölle einer Textverarbeitung oder im tiefen Schlund eines Wikis. In diesem Vortrag wollen wir uns anschauen, wie aufbauend auf bestehenden Tools und Textformaten eine möglichst redundanzfreie Dokumentation erstellt und für verschiedene Zielgruppen in ansprechenden Formaten ausgeliefert werden kann. Es wird dabei um Begriffe wie Continuous Documentation und Documentation as Code gehen.

Kontinuierliche Architekturdokumentation



Warum?



Agil?



Was?



Wie?



Kontinuierliche Architekturdokumentation



Warum?



Agil?



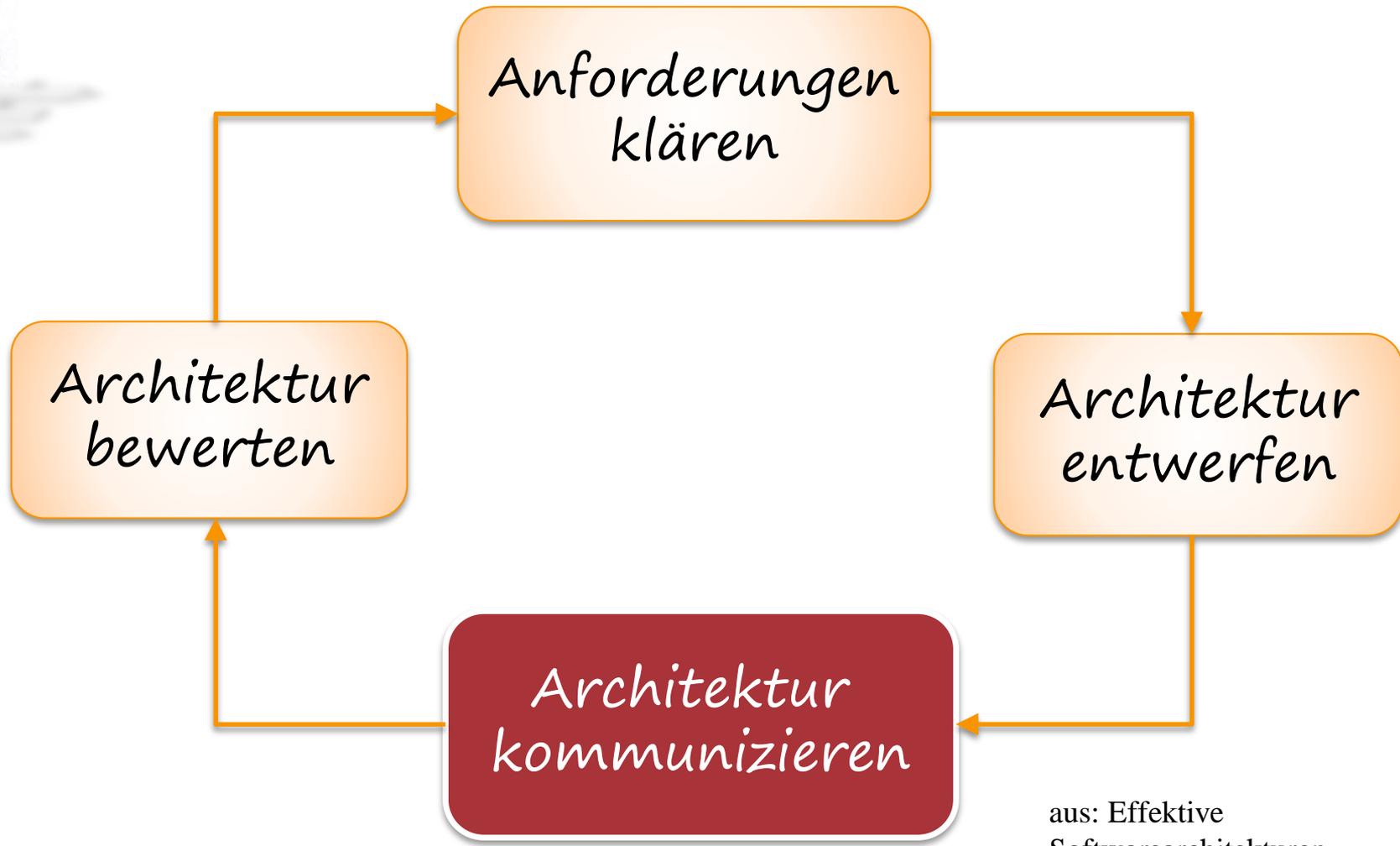
Was?



Wie?



Warum dokumentieren?



aus: Effektive
Softwarearchitekturen



Warum dokumentieren?

"In software, there is always architecture.
You can't have no architecture."

"But with it's documentation, there are two
options: you can have one or not."

"If you have one, there are two options:
the documentation matches with the real
architecture or not."



Warum dokumentieren?

"... dass auch eine **perfekte** Architektur **nutzlos** bleibt, wenn sie **nicht verstanden** wird ..."

"Die Architektur zu dokumentieren, ist der **kritische, krönende Schritt** zur **Erschaffung**."

Aus: Software Architecture Documentation in Practice von Bachmann, Bass

Interview mit Falk Sippach

“Eine perfekte Architektur ist nutzlos, wenn sie niemand versteht”

11. Februar 2016 Hartmut Schlosser

← Als Antwort an [JAXenter](#)

 **Oliver Gierke** @olivergierke · 25. Feb.
 .@JAXenter @sipsack Wie kann sie perfekt sein, wenn sie unverständlich ist? 🤔

← ↻ ❤️ 3 ⋮

 **Ralf Claussnitzer** @claussni · 25. Feb.
 @olivergierke @JAXenter @sipsack Und wie kann sie nutzlos sein, wenn sie doch gut funktioniert?

← ↻ ❤️ ⋮

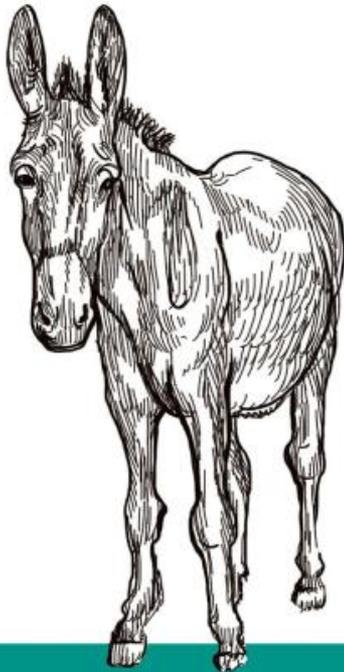
[Andere Antworten anzeigen](#)

 **Jens Schauder** @jensschauder · 25. Feb.
 @claussni @olivergierke @JAXenter @sipsack Wie kann sie funktionieren, wenn die, die sie umsetzen sollen sie nicht verstehen?

← ↻ ❤️ ⋮

 **Falk Sippach** @sipsack · 25. Feb.
 @jensschauder @claussni @olivergierke @JAXenter eine sinnvolle und

Where's the fun in just knowing what the code is supposed to do?



Essential

Excuses for Not
Writing Documentation

Entwurfsunterstützung

Kommunikation

Bewertbarkeit

Frage nach Warum

Neue Mitarbeiter

Kontinuierliche Architekturdokumentation



Warum?



Agil?



Was?



Wie?

Agil dokumentieren?

Scrum ist "murcs"
rückwärts!

"Wenn ein **Projekt**
den **Bach** runter geht,
dann nennt man das wohl
Wasserfall."



MISSVERSTÄNDNIS?

Agile Teams brauchen
nicht dokumentieren!

Endlich ...

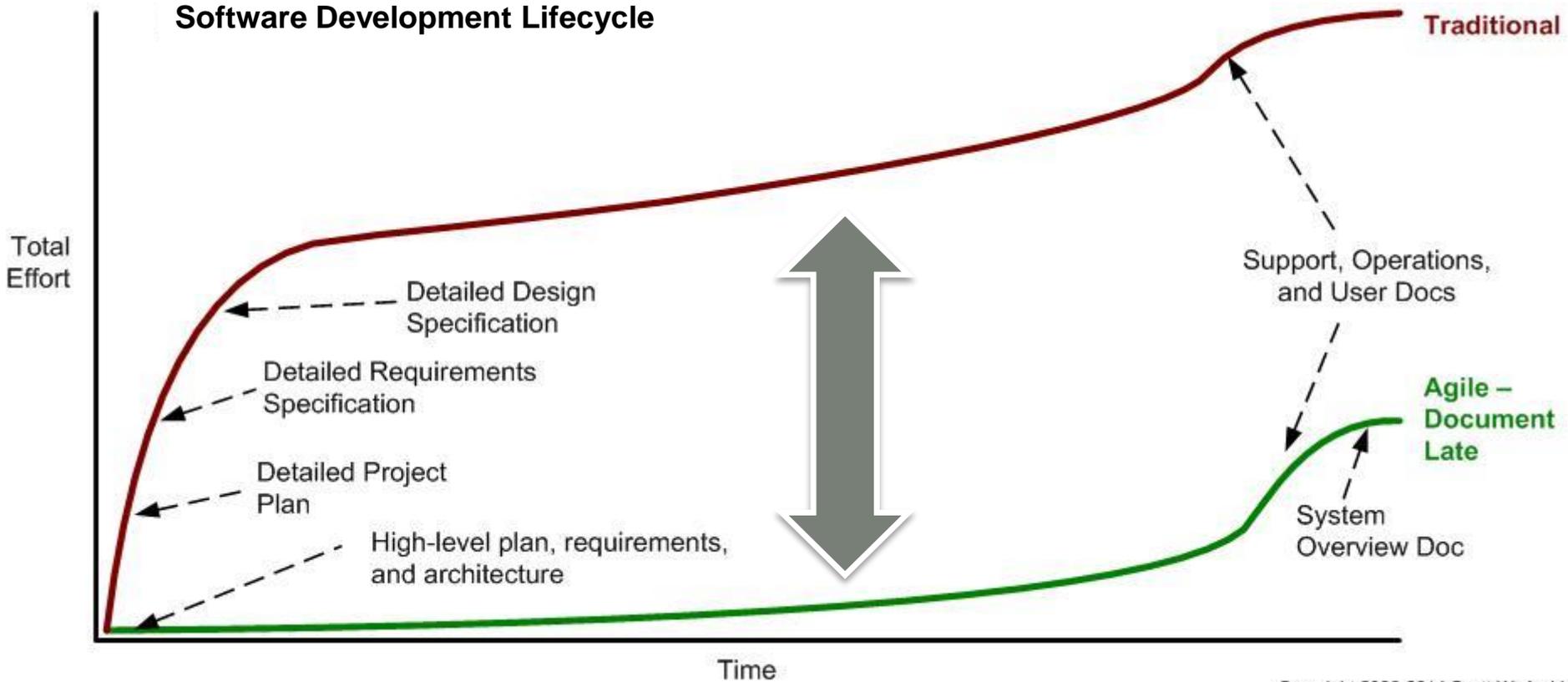
Häh?

Wer ist schuld?

**Laufende Software
wichtiger als
ausführliche Dokumentation**

Worauf das agile Manifest eigentlich abzielte ...

Documentation through the Software Development Lifecycle



Copyright 2006-2014 Scott W. Ambler

<http://www.agilemodeling.com/essays/agileDocumentationBestPractices.htm>



Agile Methoden
kennen keinen
Architekten!

**"Who needs
an Architect?"**



Architekt als Ratgeber/Mentor und Moderator



Vorbereiten
Planen
Erinnern
Delegieren
Integrieren
Prüfen

Dokumentation braucht einen

Kümmerner

Agile Projekte

iterativ
kontinuierlich

Anpassen
Reviewen
Nachbessern

Continuous Documentation



Redundanzen vermeidbar?

Quellcode verlinken
Platzhalter einbetten
Single Source of Truth

Inhalte generierbar?



WSDL, Swagger
DB-Schema
Annotationen
JavaDoc

Swagger2Markup / swagger2markup

Watch 15 Star 294 Fork 54

Code Issues 7 Pull requests 0 Pulse Graphs

A Swagger to AsciiDoc or Markdown converter to simplify the generation of an up-to-date RESTful API documentation by combining documentation that's been hand-written with auto-generated API documentation.

Spring REST Docs SUCCESS

Overview

The primary goal of this project is to make it easy to document RESTful services by combining content that's been hand-written using [Asciidoctor](#) with auto-generated examples produced with the [Spring MVC Test](#) framework. The result is intended to be an easy-to-read user guide, akin to [GitHub's API documentation](#) for example, rather than the fully automated, dense API documentation produced by tools like [Swagger](#).

JAX-RS Analyzer

Generates an overview of all JAX-RS resources in a project by bytecode analysis.

Validierung

A close-up photograph of two soldiers in dark blue uniforms with gold accents. They are wearing white gloves and aiming rifles horizontally. The soldier on the right is wearing sunglasses and a cap with a gold band. The background is a blurred outdoor setting with warm lighting.

SanityChecks
Broken Links
PDFUnit

Ausführbare Dokumentation



Ausführbare Tests
Einbettung in Dokument
Reportgenerierung

/ ALL SCENARIOS

SUMMARY

All Scenarios 3

Failed Scenarios 0

Pending Scenarios 0

TAGS

Page: Detail

Page: Start

CLASSES

▶ org

All Scenarios



✓ 3 Successful, 0 Failed, 0 Pending, 3 Total
(29.910s)

Group By ▼

Sort By ▼

Status ▼

Tags ▼

Classes ▼

✓ Detail Page Should show details page ✓ (13.411s)

Page: Detail

Given start page opened in browser (6.257s)

When click on first talk (1.143s)

Then talk details are visible (302ms)

And talk abstract is visible (21ms)

org.dukecon.server.gui.DetailPageTest

Page: Start

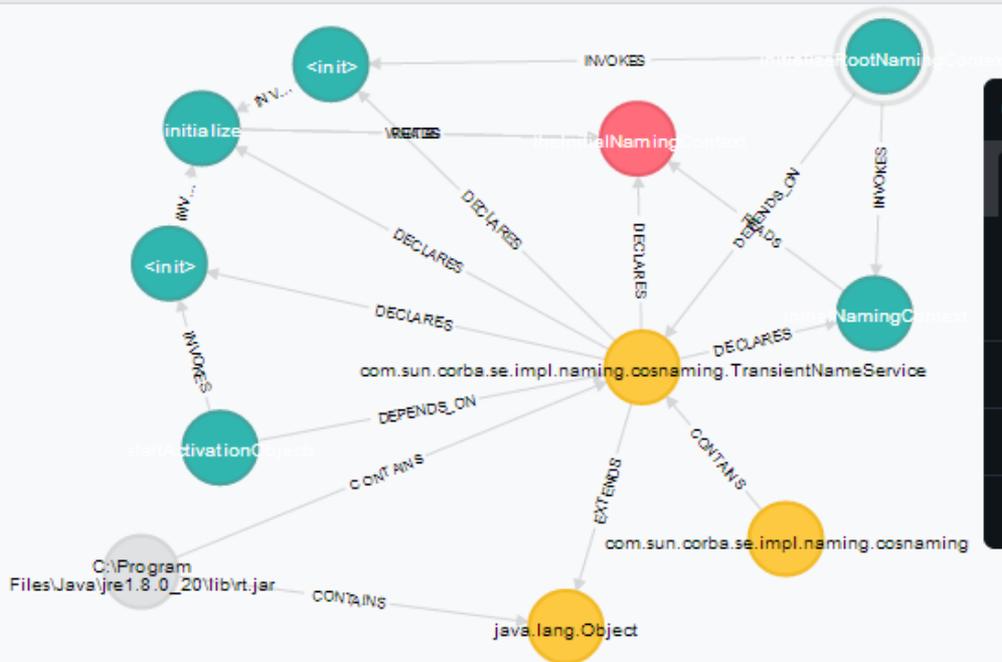
Page: Start

```
19 @Test
20 @DetailPageTag
21 public void should_show_details_page() throws InterruptedException {
22     given().start_page_opened_in_browser();
23     when().click_on_first_talk();
24     then().talk_details_are_visible()
25         .and().talk_abstract_is_visible();
26 }
```

CYPHER `match (t:Type) return t limit 1`



- Type
- Jar
- Method
- File
- Field



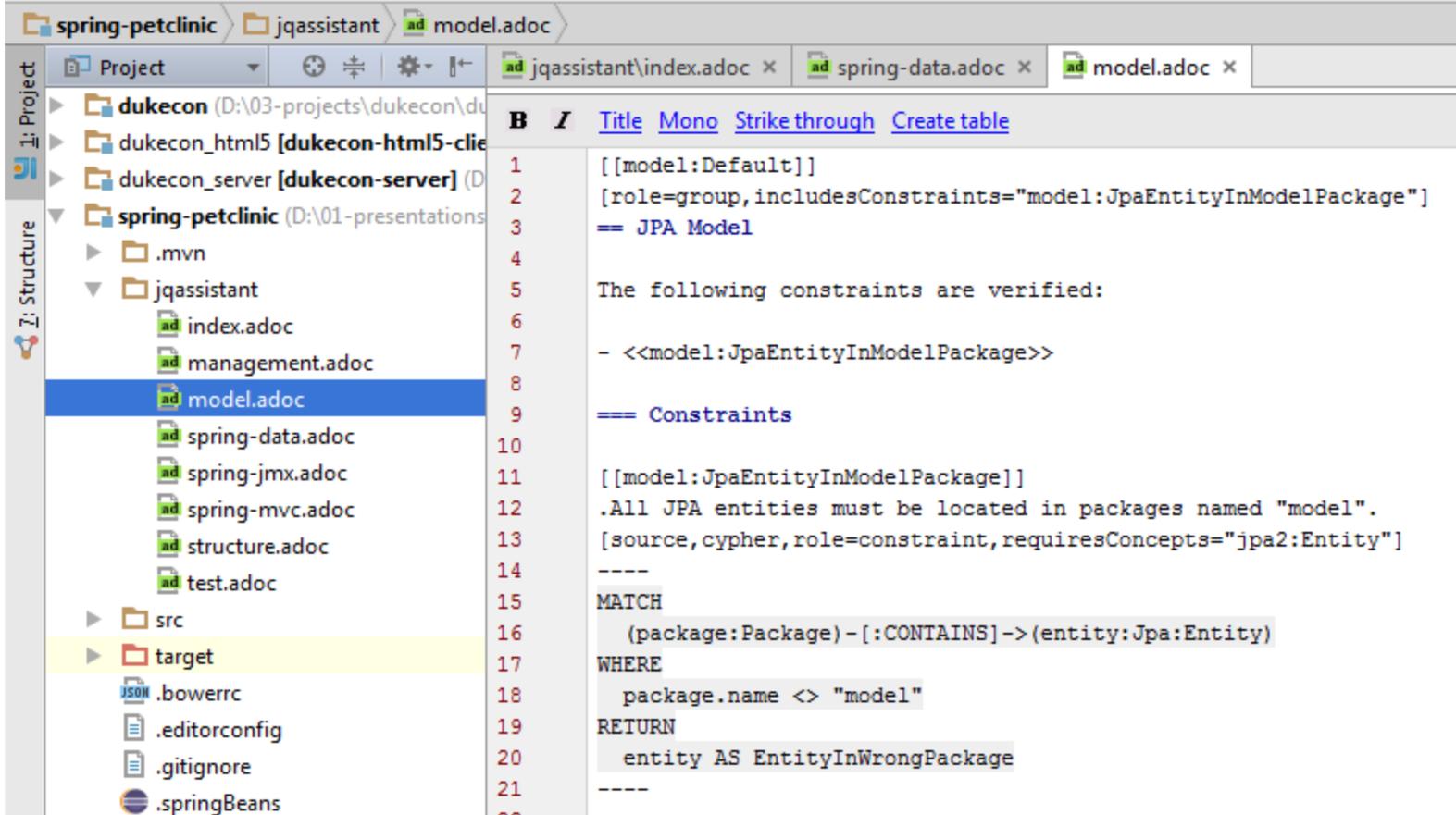
Method [20617]

Properties

signature	org.omg.CORBA.Object initializeRootNamingContext(org.omg.CORBA.ORB)
name	initializeRootNamingContext
visibility	private
static	true

✓ Displaying 11 nodes, 19 relationships





```

1  [[model:Default]]
2  [role=group,includesConstraints="model:JpaEntityInModelPackage"]
3  == JPA Model
4
5  The following constraints are verified:
6
7  - <<model:JpaEntityInModelPackage>>
8
9  === Constraints
10
11 [[model:JpaEntityInModelPackage]]
12 .All JPA entities must be located in packages named "model".
13 [source,cypher,role=constraint,requiresConcepts="jpa2:Entity"]
14 ----
15 MATCH
16     (package:Package) - [:CONTAINS]->(entity:Jpa:Entity)
17 WHERE
18     package.name <> "model"
19 RETURN
20     entity AS EntityInWrongPackage
21 ----
22
  
```

jQAssistant Report

Groups

#	Group Name	Date
1	default	2016-04-14T14:43:50
2	management:Default	2016-04-14T14:43:50
3	test:Default	2016-04-14T14:43:51
4	structure:Default	2016-04-14T14:43:52
5	model:Default	2016-04-14T14:43:55

Constraints

- Move the mouse over a constraint to view a description.
- Click on a failed constraint to open a details view.

#	Constraint Name	Count	Severity	Duration (in ms)
1	assertj:TestMethodWithoutAssertion ✓	0	info	238
2	junit4:IgnoreWithoutMessage ✓	0	info	192
3	structure:ImplementationDependencies ✓	0	info	351
4	dependency:PackageCycles ✓	0	info	178
5	structure:ControllerDependencies ✓	0	info	46
6	structure:RepositoryDependencies ✓	0	info	40
7	structure:ServiceDependencies ✓	0	info	31
8	model:JpaEntityInModelPackage ✓	0	info	42

Kontinuierliche Architekturdokumentation



Warum?



Agil?



Was?



Wie?



Was dokumentieren?

Inhalt



fundamentale

Strukturen,
Konzepte,
Entscheidungen
und Lösungsansätze

... die **wesentlich** sind, damit Systeme ihren **Anforderungen** genügen!

... die man **nicht** mehr leicht **losbekommt!**

Was gehört in die Architektur-Dokumentation?

Bausteine und Schnittstellen



Zusammenarbeit zur Laufzeit



Integration in techn. Infrastruktur



Technische Konzepte



Wichtige Entscheidungen





nur so viel wie nötig,
wenig Änderungen,
Zielgruppenbedürfnisse,
Feedback einpflegen

Pragmatik/Effektivität

Kontinuierliche Architekturdokumentation



Warum?



Agil?



Was?



Wie?



Prozess

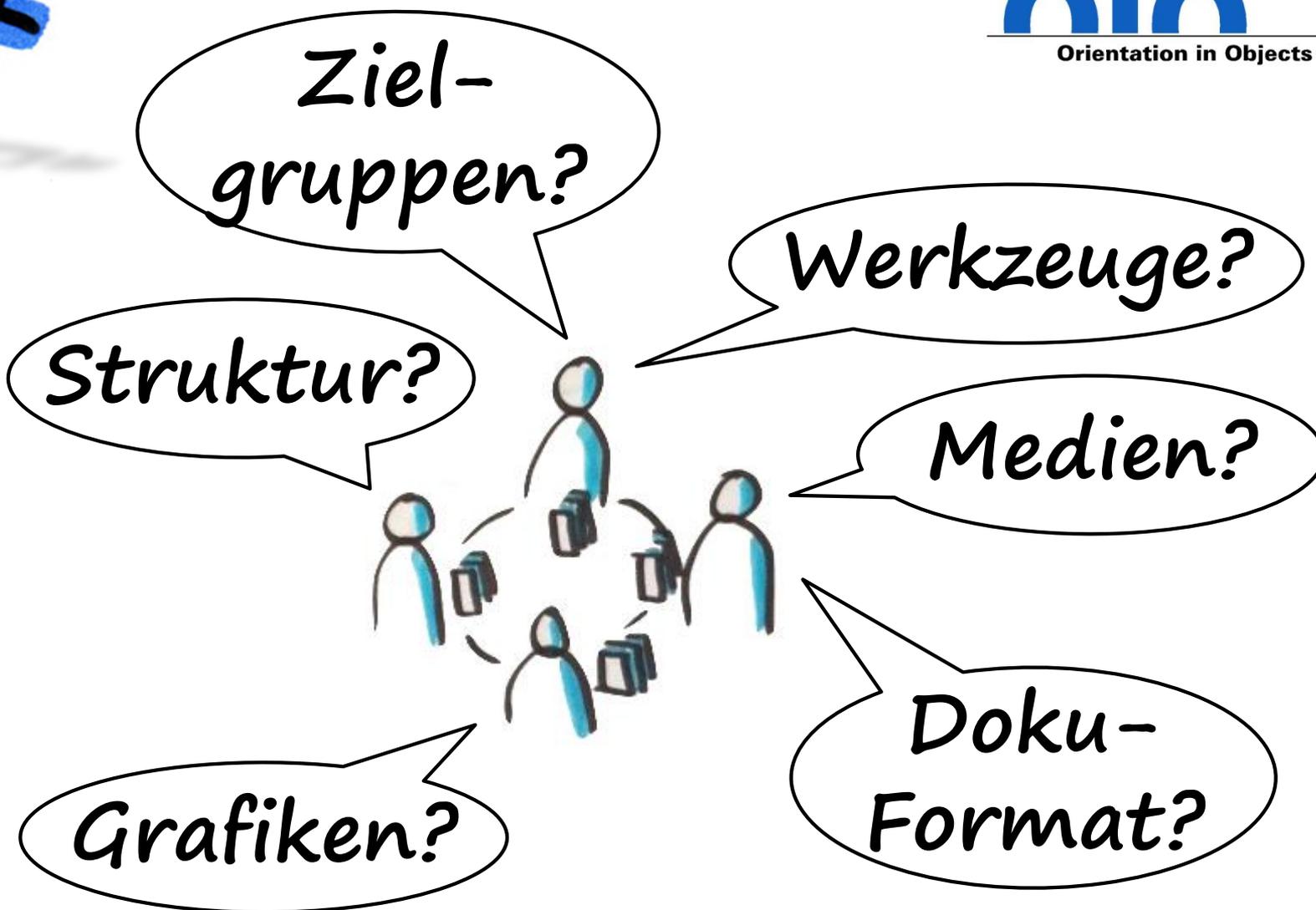
Werkzeugkette

Wie dokumentieren?



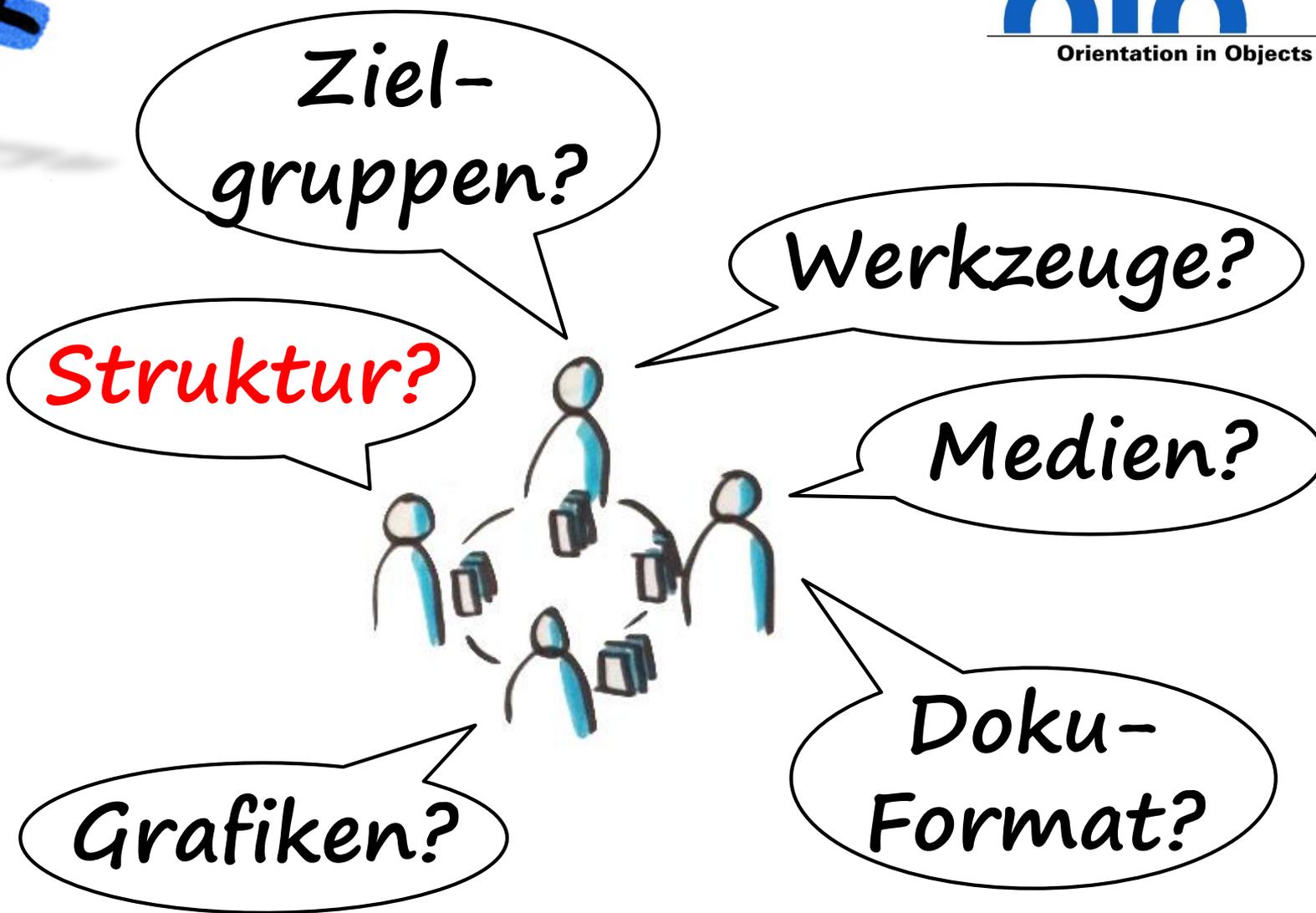


Wie dokumentieren?





Wie dokumentieren?



Zielgruppen?

Struktur?

Werkzeuge?

Medien?

Doku-Format?

Grafiken?

1. Einführung und Ziele

Aufgabenstellung, Qualitätsziele, eine Kurzfassung der architekturrelevanten Anforderungen (insb. die nichtfunktionalen), Stakeholder.

2. Randbedingungen

Welche Leitplanken schränken die Entwurfsentscheidungen ein?

3. Kontextabgrenzung

In welchem fachlichen und/oder technischen Umfeld arbeitet das System?

4. Lösungsstrategie

Wie funktioniert die Lösung? Was sind die fundamentalen Lösungsansätze?

5. Bausteinsicht

Die statische Struktur des Systems und die Implementierungsteile.

6. Laufzeit

Zusätzliche Bausteine zur Laufzeit, gezeigt an Beispielen ("Szenarien")

7. Verteilungssicht

Deployment: Auf welcher Hardware werden die Bausteine betrieben?

8. Querschnittliche Konzepte und Muster

Wiederkehrende Muster und Strukturen
Fachliche Strukturen.
Querschnittliche, übergreifende Konzepte, Konfigurations- oder Einsatzanleitungen, die über mehrere Ebenen hinweg
projekt-/systemübergreifend wendbar!

9. Entscheidungsfindungen

Entscheidungen und wichtige Entscheidungen.

10. Qualitätsszenarien

Qualitätsbaum sowie dessen Konkretisierung durch Szenarien

11. Risiken

12. Glossar

Wichtige Begriffe.

anforderungsbezogene Informationen

Strukturen der Lösung (Sichten)

übergreifende (technische) Informationen

besonders wichtige Entscheidungen

Einheitliche Dokumentenstruktur, z. B. arc42

arc42 Templates

	Online / Wiki	Version	Download (via dropbox)
	arc42 template in English	6.1	<ul style="list-style-type: none"> • docx • The template as a Confluence-space: <ul style="list-style-type: none"> • for Confluence version 5.x or 6.x • for confluence version 4.3 • Enterprise-Architect (tm) Model (EAP), slim version. • AsciiDoc version (via GitHub) <p>We created an introductory example (HtmlSanityChecker) to give an impression how arc42 could look like ...</p>
	arc42 Template auf Deutsch	6.0	<ul style="list-style-type: none"> • docx • Beispiele für Qualitätsanforderungen ("Q-Szenarien") • Das Template als Confluence-Space: <ul style="list-style-type: none"> • für Confluence 5.x und 6.x • für Confluence Version 4.3.x • Enterprise-Architect (tm) Model (EAP) (schlanke Version.) • xmind-Version (Mindmap, to be published soon)
	arc42 Template Deutsch	7.0	<p>update currently under internal review. Structure will remain unchanged, numerous corrections to help text. To be released Jan/Feb 2016</p> <ul style="list-style-type: none"> • AsciiDoc (master version) • docx • xmind (Mindmap) • Confluence
	arc42 template in Spanish work in progress, Microsoft Word version currently completed.	5.0	<ul style="list-style-type: none"> • docx-Version



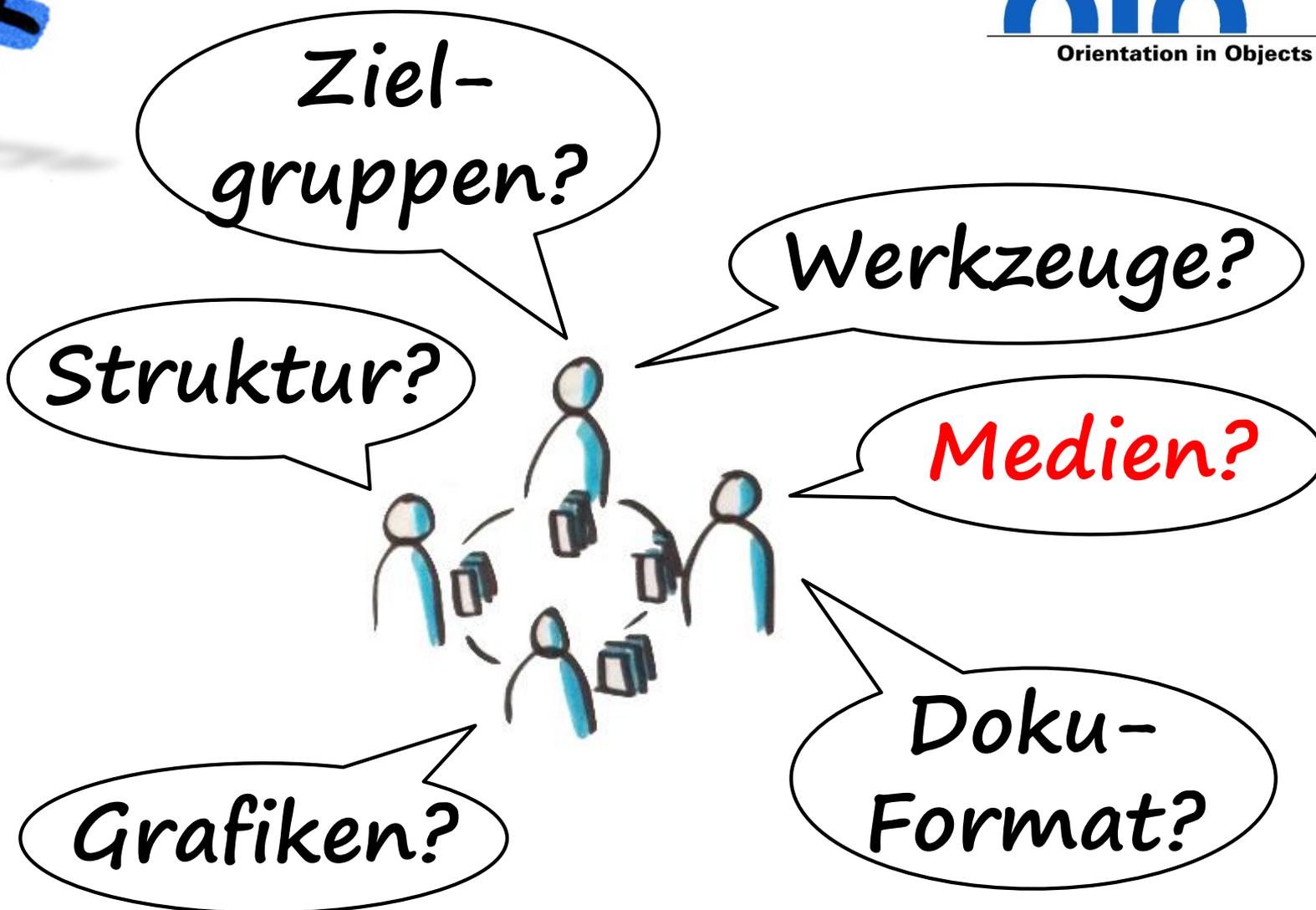
Confluence

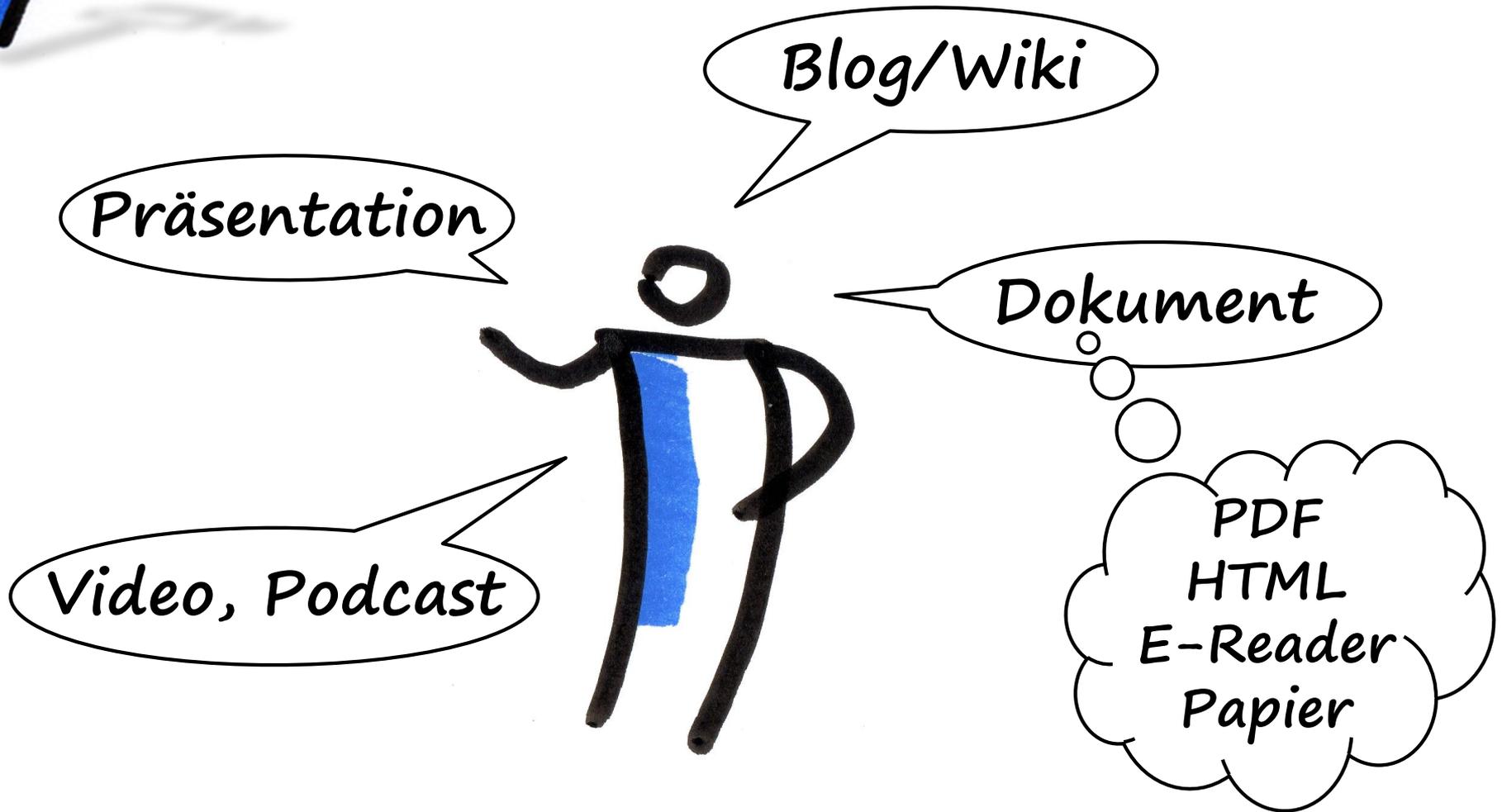


Asciidoctor



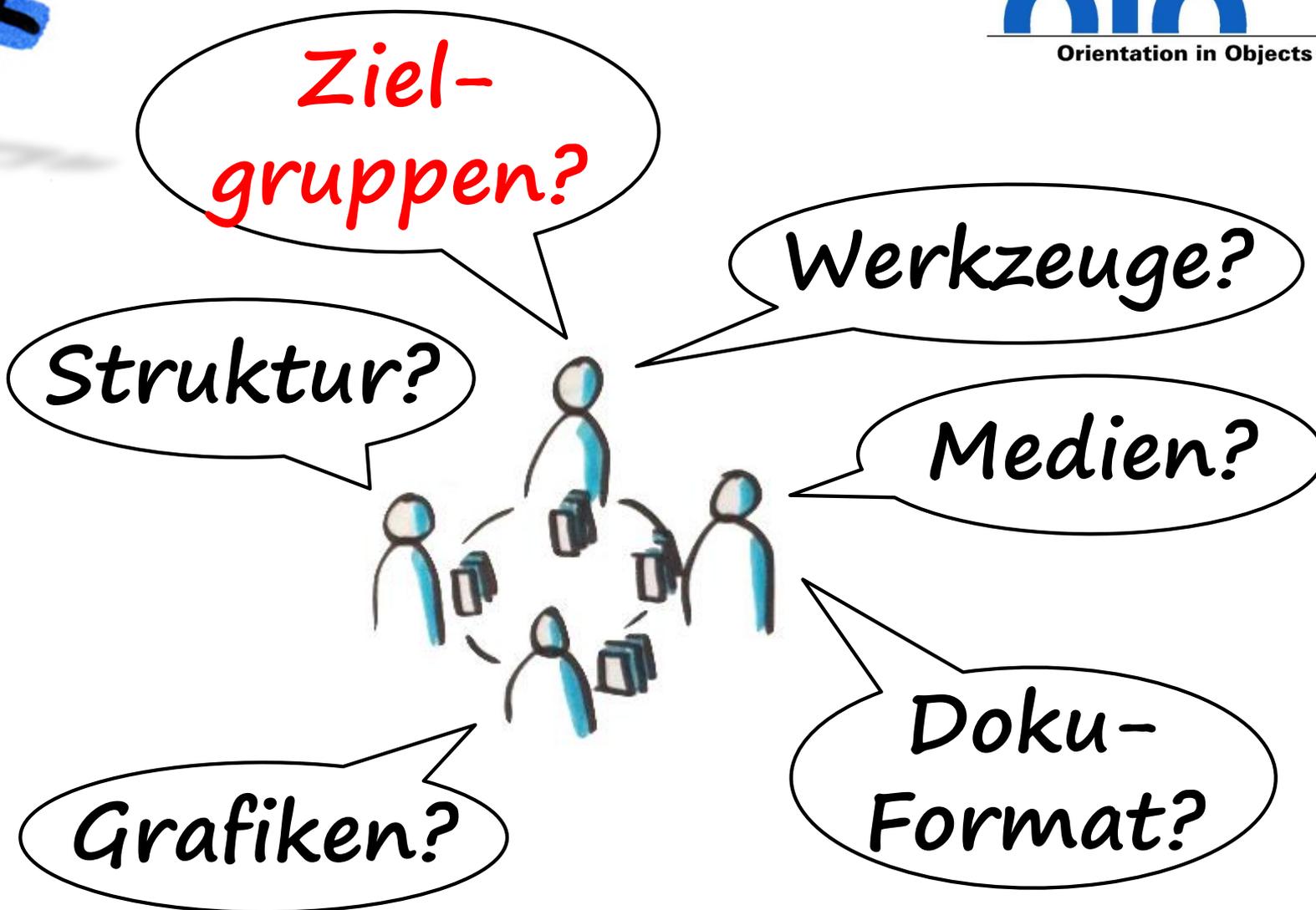
Wie dokumentieren?



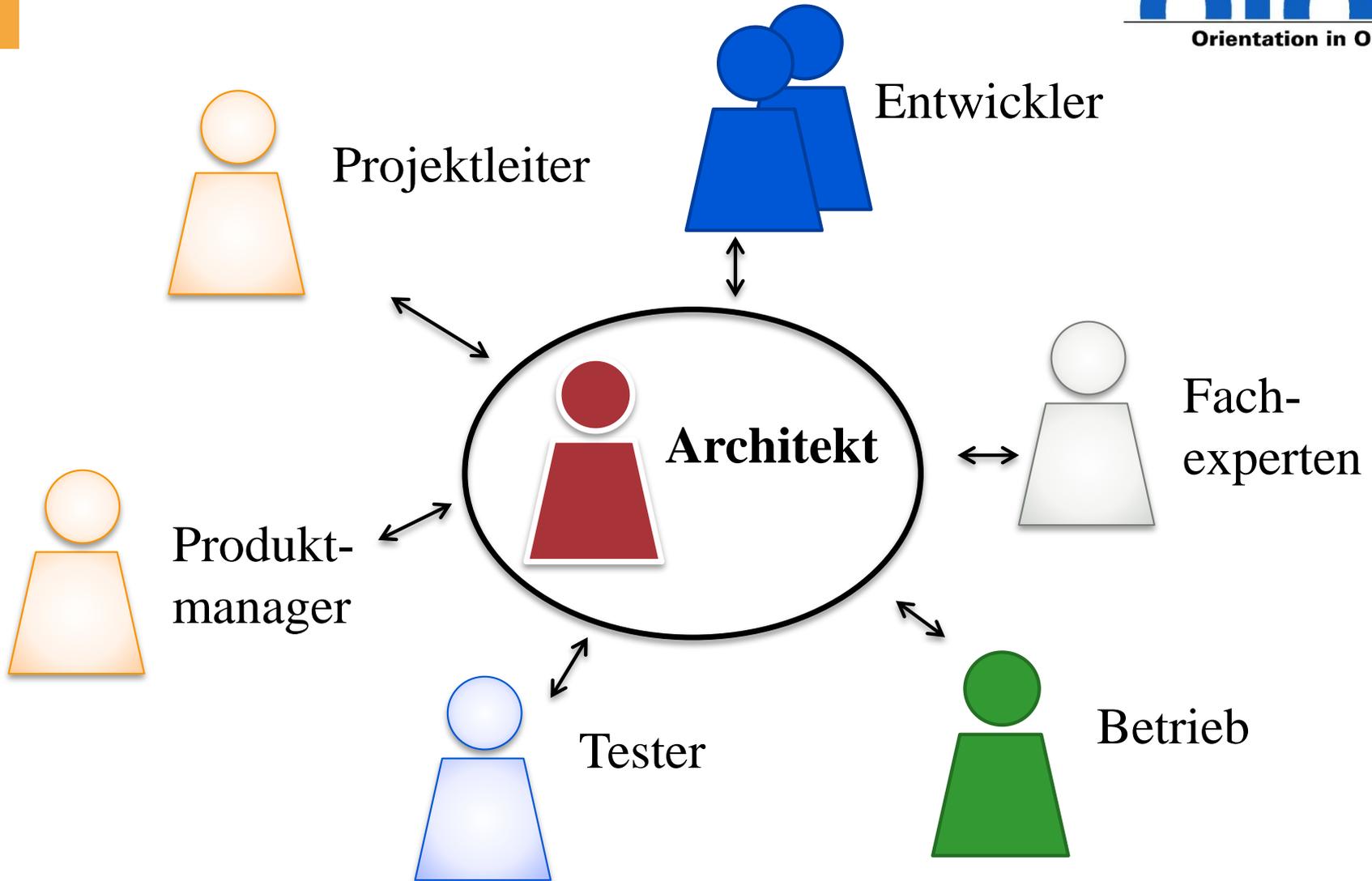




Wie dokumentieren?

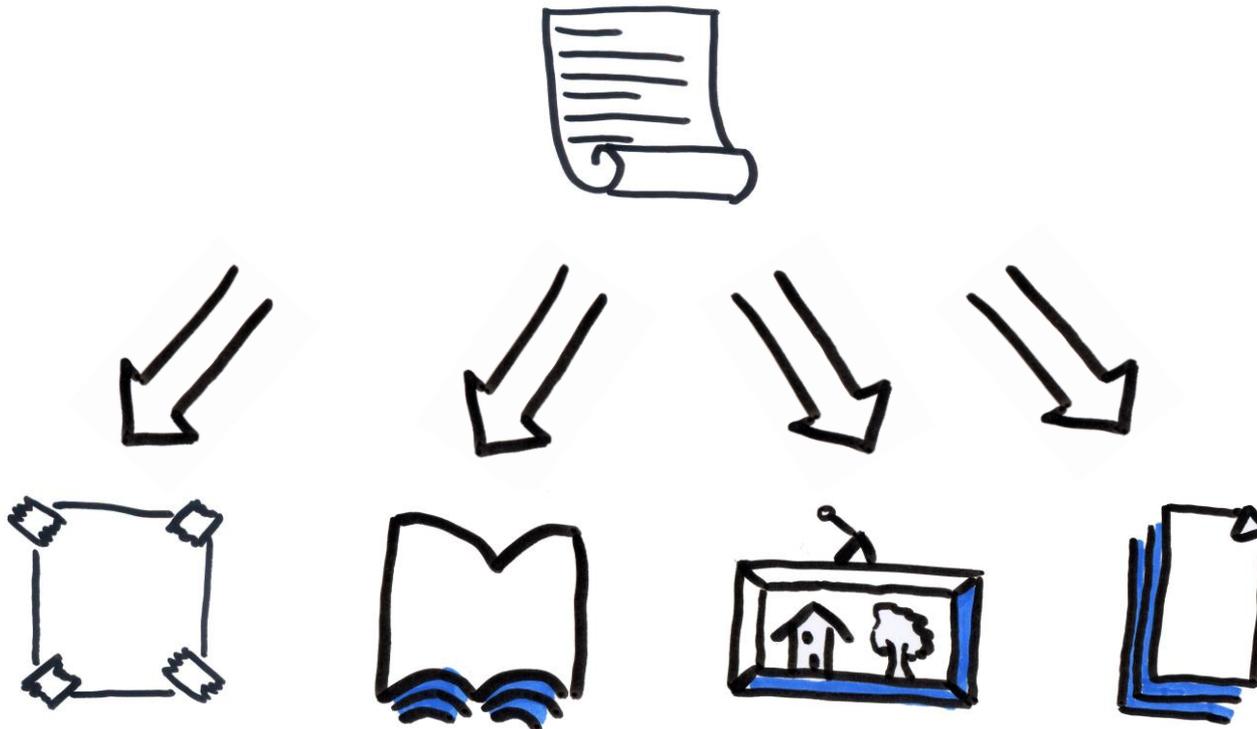


Zielgruppen identifizieren



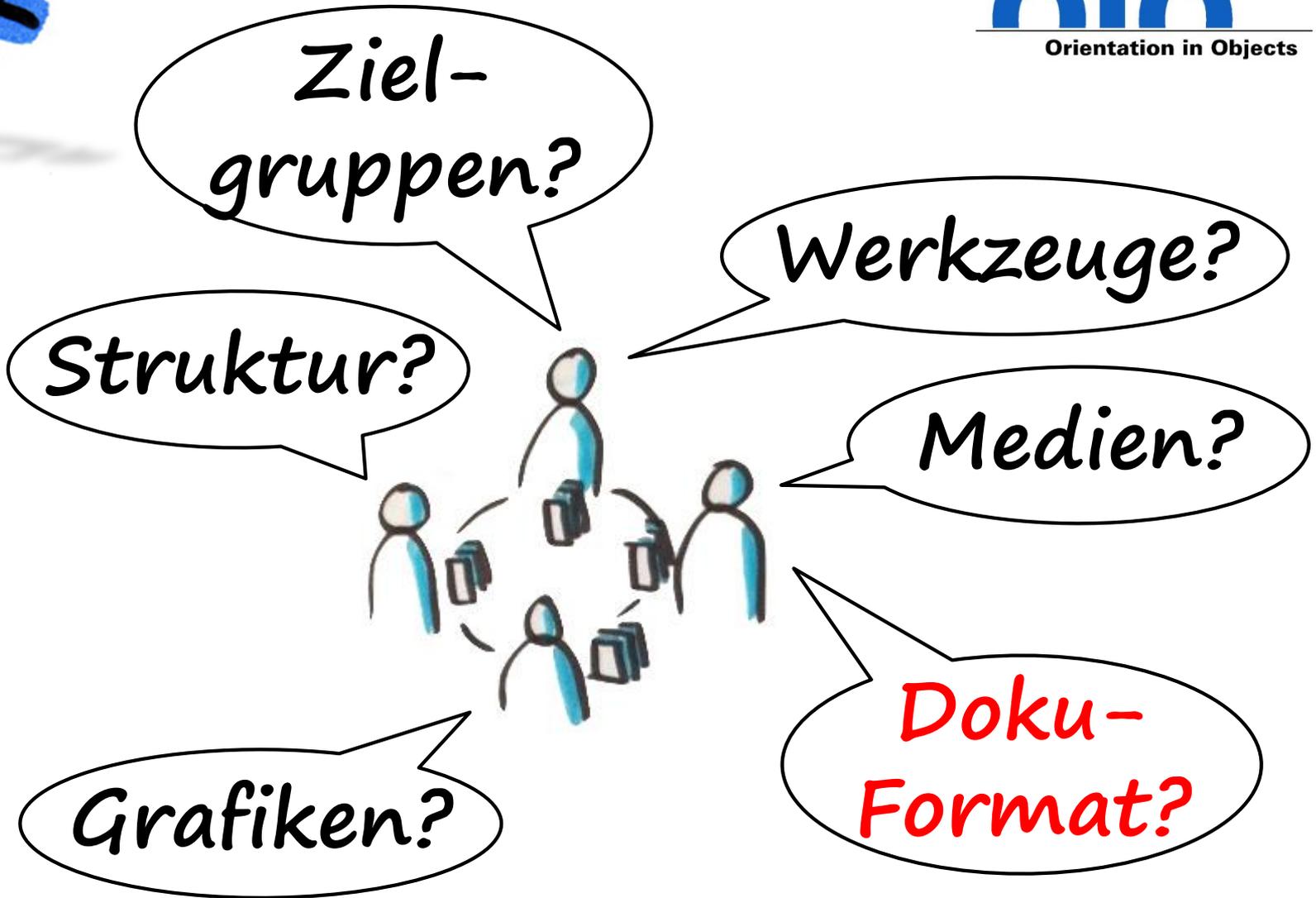
Braucht jeder alles?

- Inhalte pro Zielgruppe festlegen
- möglichst ein zentrales Dokument
- (automatisiert) verschiedene Ausgabedokumente generieren





Wie dokumentieren?



Zielgruppen?

Struktur?

Werkzeuge?

Medien?

Grafiken?

Doku-
Format?

Hauptsache,
du machst es
nicht mit Word!



Ich mach's mit Latex.

Kondome schützen.
Für eine Nacht oder mehr, mach's mit.

GIB AIDS KEINE CHANCE

STI
Sexuell übertragbare Infektionen

www.sti-hilfe.de

Unser täglich Entwickler-Brot:

- Plain-Text
- Entwicklungsumgebung
- Kommandozeilenwerkzeuge
- Versionsverwaltung

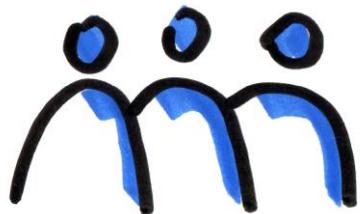




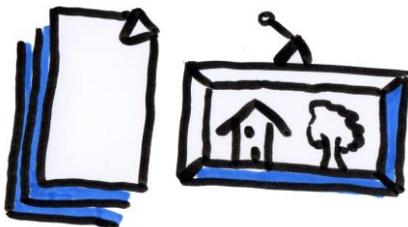
Code-Nähe
Ablage im Repo
Versionier-/diffbar
Synchrone Auslieferung
Offlinefähig
Teil des Build-Prozess
Generierung
Automatisierung
Flexible Ausgabeformate

Documentation as Code

Workflow



Entwickler



 **Bitbucket**  **GitHub**



Maven




HTML
PDF



Leser

Plain-Text, leicht lesbar, einfach editierbar, automatisiert verarbeitbar

Plain-Text

kollaborativ

Wikis

eingeschränkte Lesbarkeit

LaTeX/DocBook

kurz & knapp

Austauschformat

Mindmaps

Richtext

Normaler Text wird so dargestellt wie eingegeben.

`*Kursiv*`, `**Fett**` und `***Fett kursiv***`
und `___Fett kursiv___`

Markiert Text als ``Inline-Quelltext``

Ein Code-Block
durch Einrückung
mit vier Leerzeichen

`* Ein Punkt in einer ungeordneten Liste`
`* Ein Unterpunkt, um vier Leerzeichen eingerückt`

`1. Ein Punkt in einer geordneten Liste`
`2. Ein weiterer Punkt`
`1. Noch ein Punkt bei mehrfacher Angabe derselben Ziffer`

`# Überschrift in Ebene 1`
`#### Überschrift in Ebene 4`

Normaler Text wird so dargestellt wie eingegeben.

Kursiv, **Fett** und ***Fett kursiv*** bzw. *Kursiv*, **Fett** und ***Fett kursiv***

Markiert Text als `Inline-Quelltext`

```
Ein Code-Block
durch Einrückung
mit vier Leerzeichen
```

- Ein Punkt in einer ungeordneten Liste
 - Ein Unterpunkt, um vier Leerzeichen eingerückt
- 1. Ein Punkt in einer geordneten Liste
- 2. Ein weiterer Punkt
- 3. Noch ein Punkt bei mehrfacher Angabe derselben Ziffer

Überschrift in Ebene 1

Überschrift in Ebene 4

```
= AsciiDoc is Writing Zen
Doc Writer <doc.writer@example.com>
:icons: font
```

```
_Zen_ in the *art* of writing `plain text` with
http://asciidoc.org[AsciiDoc].
```

```
[TIP]
```

```
Use http://asciidoc.org[AsciiDoc] for the best AsciiDoc
experience.footnote:[Not to mention the best looking output!]
Then icon:twitter[role=aqua] about it!
```

```
== Sample Section
```

```
[square]
```

- * item 1
- * item 2

```
[source,ruby]
```

```
----
puts "Hello, World!"
----
```

AsciiDoc is Writing Zen

Doc Writer – doc.writer@example.com

Zen in the art of writing plain text with [AsciiDoc](#).



Use [AsciiDoctor](#) for the best AsciiDoc experience.^[1] Then  about it!

Sample Section

- item 1
- item 2

```
puts "Hello, World!"
```

1. Not to mention the best looking output!



Michael Simons
@rotnroll666



Folge ich

Think I'm more a Markdown person than AsciiDoc. The results are great, but Markdown needs less concentration to write.



Ralf D. Müller
@RalfDMueller



Folgen

@rotnroll666 but the possibilities of AsciiDoc are better! Include code directly from the repository, render plantUML, subdocuments etc...

- JavaDoc/Annotations
- Enumwerte/Zustandsübergänge (Enum-Werte)
- Schnittstellenbeschreibung (WSDL, Swagger, WADL)
- DB-Schema (SchemaCrawler)



DOCUMENT NAME

WORDS: 485

document.md

MARKDOWN



PREVIEW

1 ▾ # Dillinger

2

3 Dillinger is a cloud-enabled, mobile-ready,
offline-storage, AngularJS powered HTML5 Markdown
editor.

4

5 - *Type some Markdown on the left*

6 - *See HTML in the right*

7 *Magic*



Dillinger

Dillinger is a cloud-enabled, mobile-ready, offline-storage,
AngularJS powered HTML5 Markdown editor.

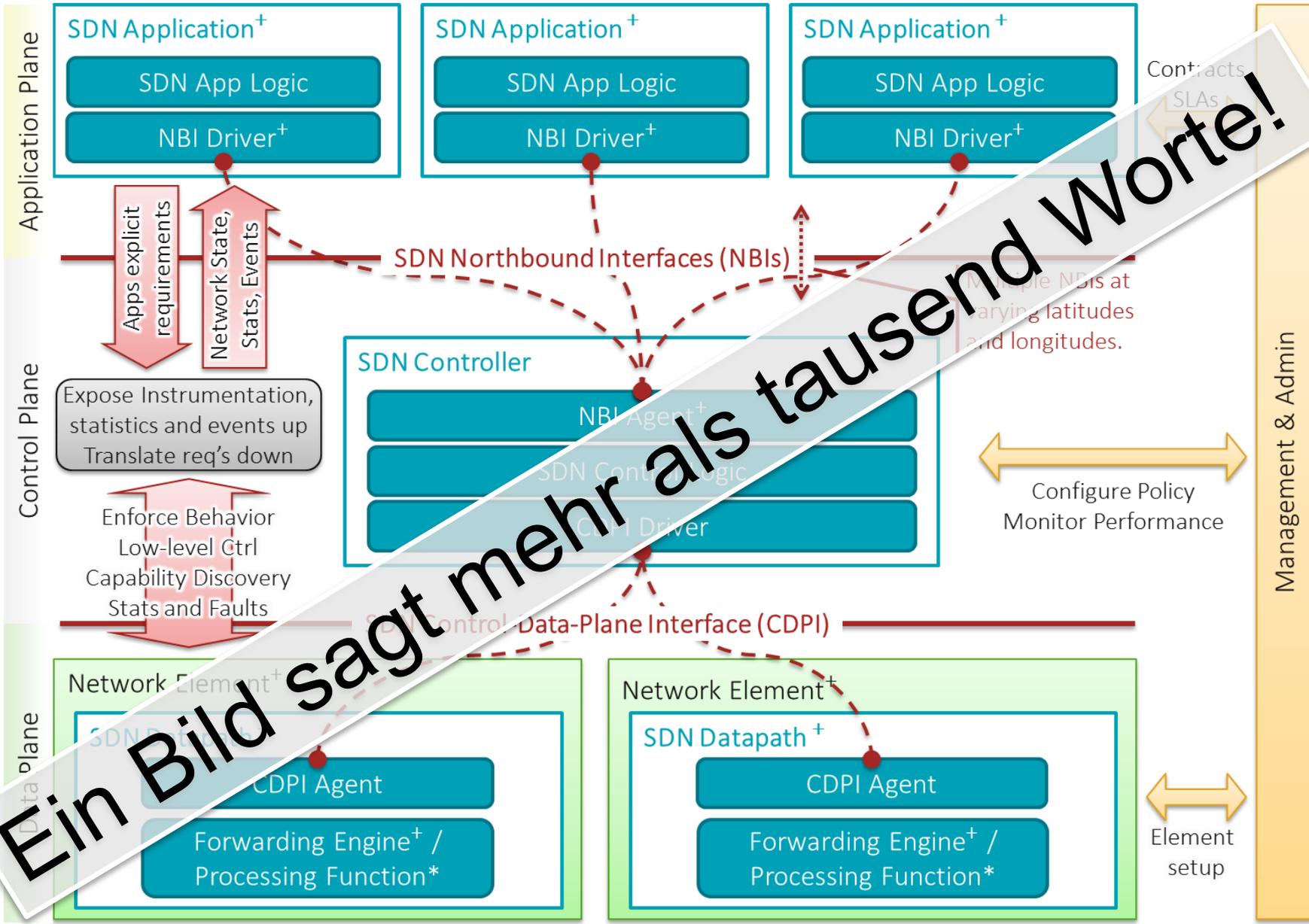
- Type some Markdown on the left
- See HTML in the right
- Magic

... und Sublime, Atom, IntelliJ IDEA, Eclipse, ...



Wie dokumentieren?





The screenshot displays the yEd software interface. The main workspace shows a network diagram with a person icon, a laptop, a server rack, and a database cylinder, all connected by arrows. The interface includes a menu bar (Datei, Bearbeiten, Ansicht, Layout, Werkzeuge, Gruppierung, Fenster, Hilfe), a toolbar with various icons, and several panels: 'Übersicht' (Overview), 'Nachbarschaft' (Neighborhood), 'Struktur' (Structure), 'Palette' (Palette), and 'Eigenschaften' (Properties). The 'Eigenschaften' panel shows a table with the following data:

Allgemein	
Knotenzahl	4
Kantenzahl	3
Daten	
Beschreibung	

Text overlay: yEd ist ein kostenloses Visualisierungsprogramm

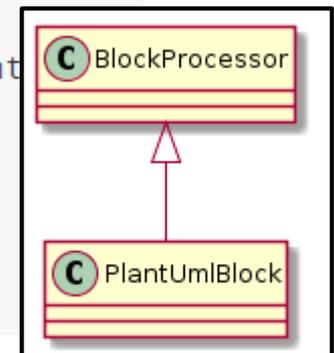
... und Enterprise Architect, Magic Draw, ...

- UML-Modelle
- Bildverarbeitung/Visualisierungsprogramme

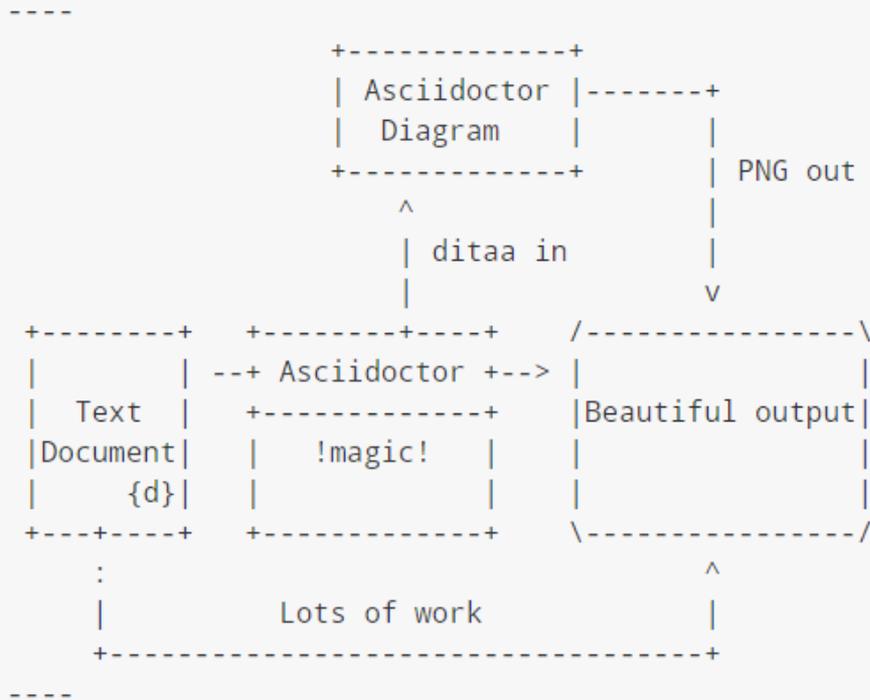
```
![Alternativtext] (bild.png "Bildtitel hier")  
image::bild.png[Alternativtext]
```

- Textuelle Beschreibung (ASCII-Art)

```
[[main-classes]]  
.The PlantUML block extension class  
[plantuml, sample-plantuml-diagram, alt="Class diagram", width=135, height=100]  
----  
class BlockProcessor  
class PlantUmlBlock  
BlockProcessor <|-- PlantUmlBlock  
----
```

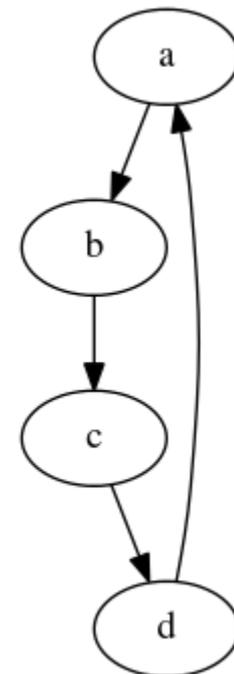
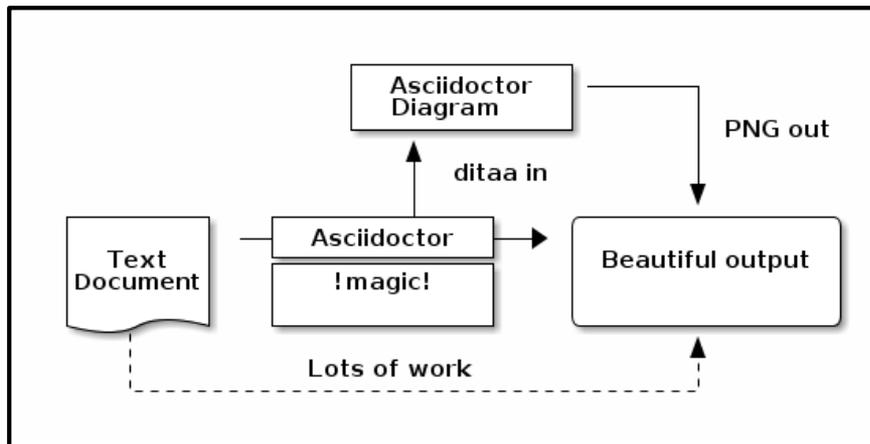


[ditaa]



[graphviz, dot-example, svg]

```
-----  
digraph g {  
    a -> b  
    b -> c  
    c -> d  
    d -> a  
}  
-----
```



Generation Plain-Text Diagramme

Quellen:

Sourcecode
DB-Schema
XML-Modell



Muss nicht aktuell gehalten werden!

```

> A: DoWork;
activate A;

A ->> B: Create Request;
activate B;

B ->> C: DoWork;
activate C;

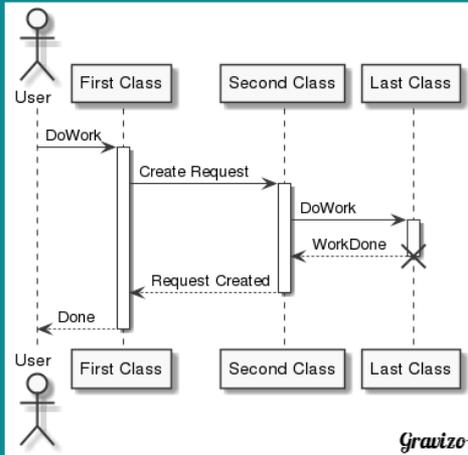
C -->> B: WorkDone;
destroy C;

B -->> A: Request Created;
deactivate B;

A -->> User: Done;
deactivate A;

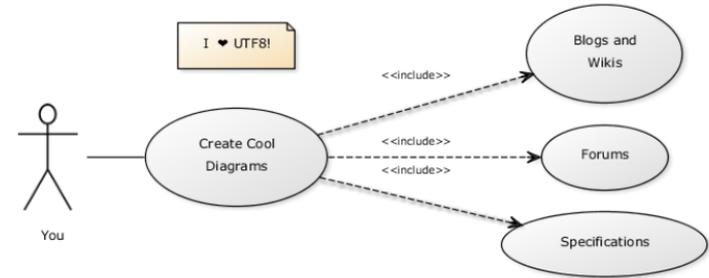
@enduml
'>

```




beta!
 Made by pocketworks mobile
 Enterprise mobile and web solutions
yUML
[HOME](#) [PRO-ACCOUNT](#) [SELF-HOSTING](#) [FAQ](#) [TOOLS](#) [FORUM](#) [TWITTER](#) [BLOG](#)

Create and share simple UML diagrams in your blogs, wikis, forums and emails.



PlantText - The expert's design tool

- Home
- Editor
- Feedback
- About
- Blog

Select a sample... ▾

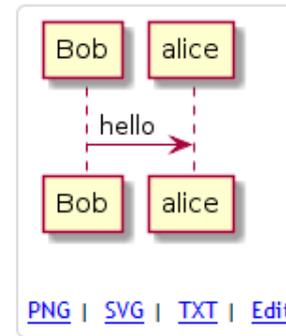


Hi! Please login -> [Login / Register](#)

Refresh
Save
Load

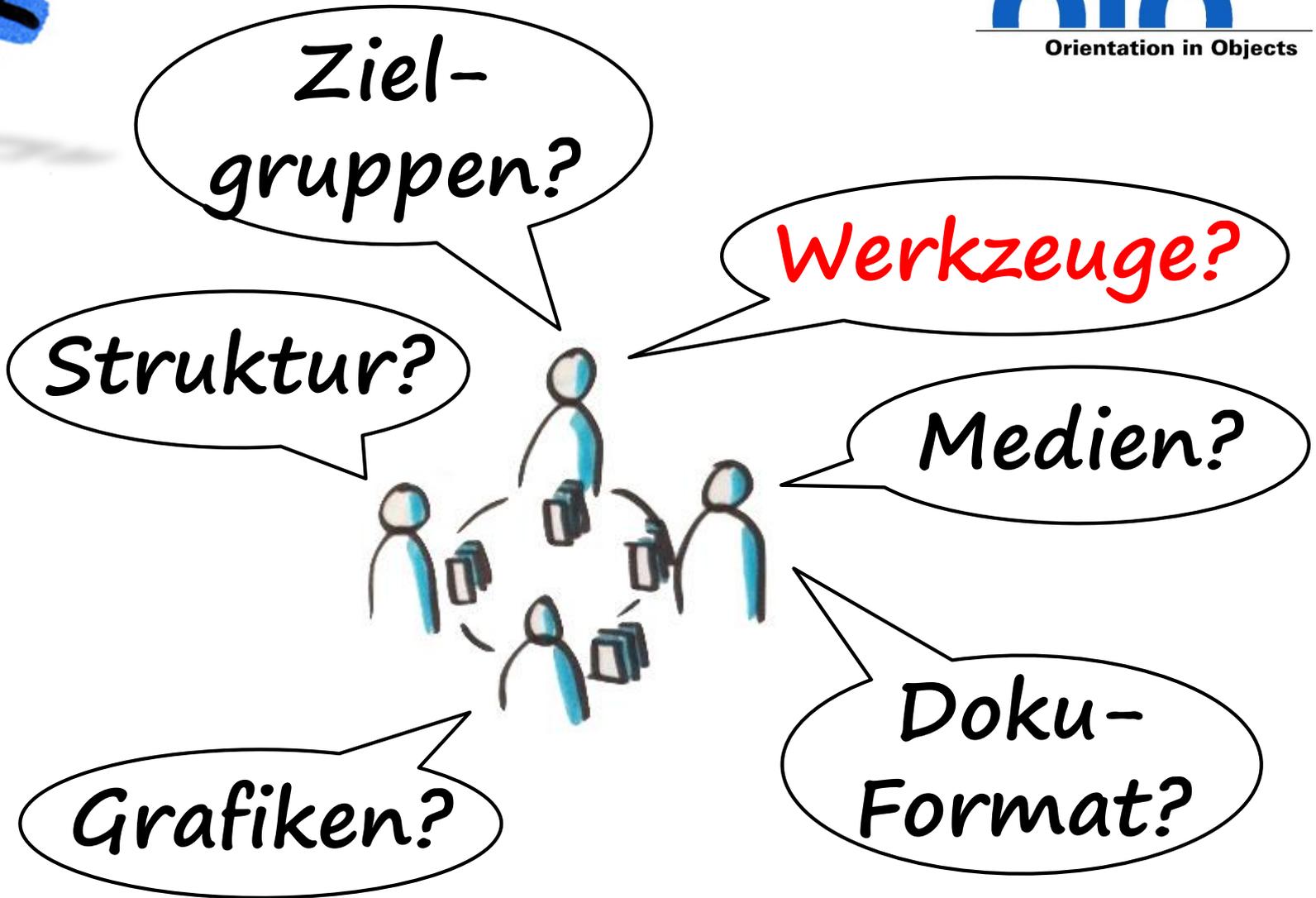
```
1 Bob -> alice: hello
```

Refresh
Save
Load





Wie dokumentieren?



The image shows three large, white icebergs floating in a deep blue ocean. The sky is a clear, bright blue. The icebergs have various shapes and textures, with some showing signs of melting and smaller ice chunks floating around them. The water is calm, reflecting the light from the sky and the ice.

Verschiedene Szenarien

- Schreiben in Markdown in IDE (IntelliJ IDEA) inkl. Preview
- PDF-Erzeugung mit PanDoc über LaTeX-Zwischenschritt inkl. Corporate Design
- UML-Diagramme mit PlantUML, Live-Ansicht/Export aus IDE
- andere Diagramme mit yEd, Export als *.png
- Stash/Bitbucket Server als Repo
 - rendert Markdown direkt in Weboberfläche
 - readme.md Verlinkungen auf wichtige Dokumente

"... fast alles ist leicht versionier- und diffbar"

"Generiertes PDF stellt alles bisherige in den Schatten."

"Entwickler finden es klasse, Leser innerhalb der Firma: finden das generierte PDF sehr gut und hübsch."

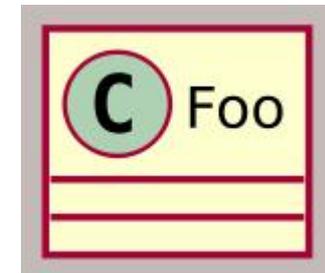
"... jeder Entwickler kann ändern..."

**"Nie wieder anders!
Ich bin voll überzeugt."**



Szenario 2: AsciiDoctor, Maven, PlantUML

- Erstellen AsciiDoc und PlantUML in IDE mit Preview
- Maven-Plugin zum Erzeugen des HTML/PDF



AsciiDoctor

Maven[™]

- Quellcode parsen
 - Reflection
 - Spring Kontext
 - ...
- in Unit-Test aus Klassen-Strukturen Diagramm-Markup erzeugen
 - z. B. PlantUML
 - als Text-Datei ablegen und in Markup-Dokumentation verlinken
- im Build-Prozess als Input für Markup-Konvertierung einlesen

Szenario 4: Schnittstellenbeschreibung

- Generierung aus WSDL, WADL, Swagger
- Einbindung in Build-Prozess
- Swagger2Markup
- JAX-RS Analyzer
- Spring REST Docs

Asciidoctor

MavenTM



Szenario 5: Ausführbare Dokumentation

- Quellcode-Struktur in Graph-Datenbank importieren
- Architektur-Regeln als Graph-Abfragen in AsciiDoc einbetten



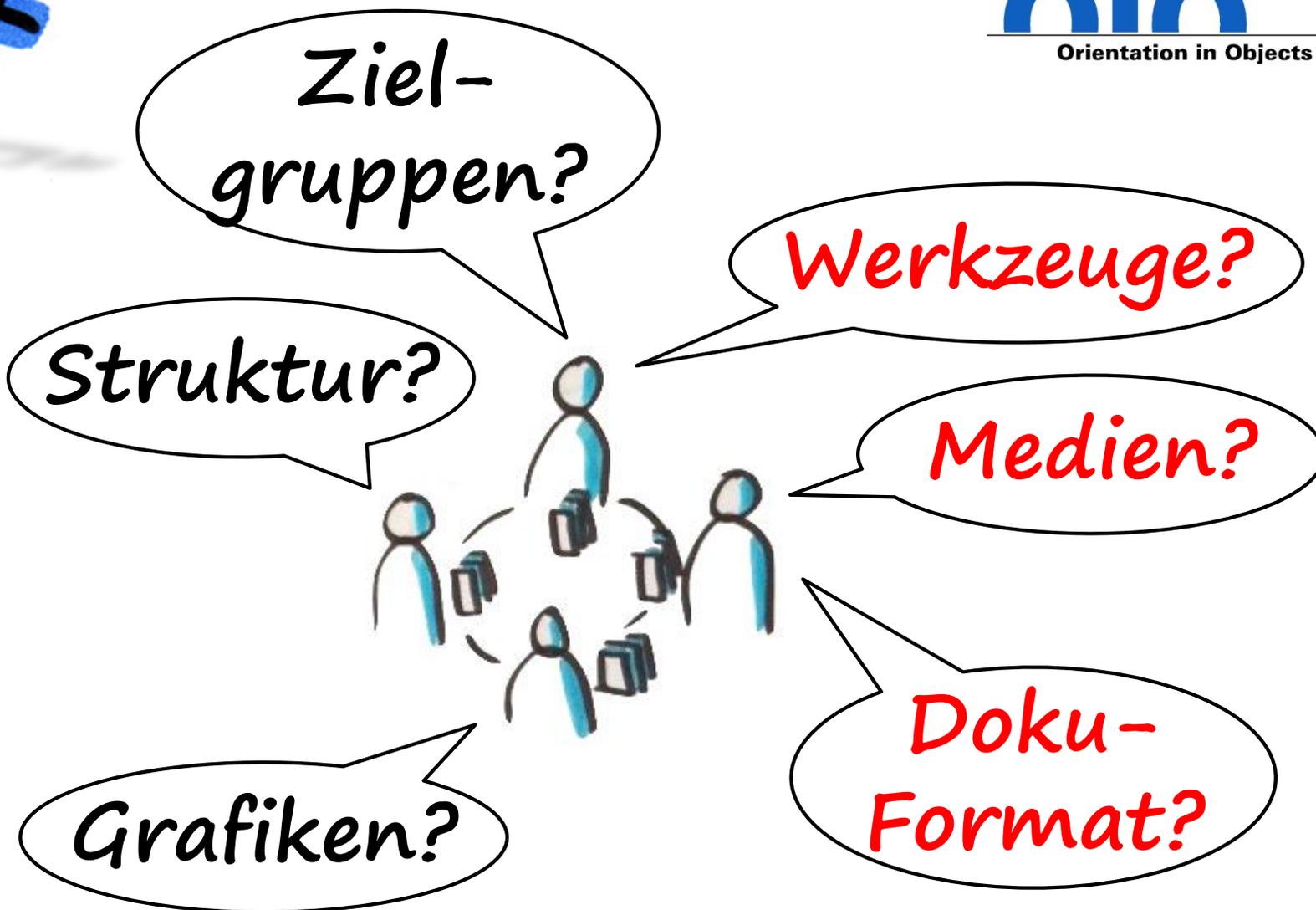
jQAAssistant

AsciiDoctor

Maven™



Wie dokumentieren?



A close-up portrait of a middle-aged man with a friendly expression. He has short, slightly messy grey hair, a full grey beard and mustache, and is wearing round, thin-rimmed glasses. He is dressed in a red t-shirt under a grey zip-up jacket. The background is a soft, out-of-focus green, suggesting an outdoor setting with foliage.

Wiki

Zusammenarbeit



Verlinkung
Review-/Redaktions-Prozess
Prozess-Unterstützung
Abbildung Workflow
Erweiterung über Plugins
Alles in einer Box!



Strukturiert?
Plain-Text?
Offlinefähigkeit?
Versionierung?
Code-Nähe?
Automatisierung?
Druckausgabe?
Zielgruppen?
Kontextwechsel

Schlund des Wiki

Open the workbox

Personal tasks

Choose Tasks

Add personal task

Add a task Add task

Blog about the changes to the logo Rach Admin

Which nerd are you?

Blog about chocolate

Buy chocolates

View tasks details

Tasks

Mark task as complete

Drag and drop to prioritise

View completed View completed tasks

Paragraph

B I U A A

Check this out @je

User suggestions

Jens Schumacher

Jeff Daniels

Jerry Seinfeld

Mentions

Confluence

Unlike You and 4 others like this

No labels

6 Comments

Matt

Really loving the new icons and font, though I am kind of curious as to why Charlie's leg seems to have been cut off 😊

How long until we update the logo stuck to the outside of CornX? 😊

Reply • Edit • Delete • Unlike • You and Jessie like this • Feb 15, 2013

Jerry AUTHOR

Hrm... I'm not sure about the CornX logo. There's a good chance I'd guess that it won't be updated due to the likely move timetable, the time to get it changed (given Sydney City Council, the Heritage Council etc all have to be involved.)

Reply • Edit • Delete • Like • Feb 15, 2013

New comment from Ryan Show Ignore

Kommentare

XConfluence Spaces People Calendars Create

6 JIRA links

JIRA links

Sprints

Sprint 5

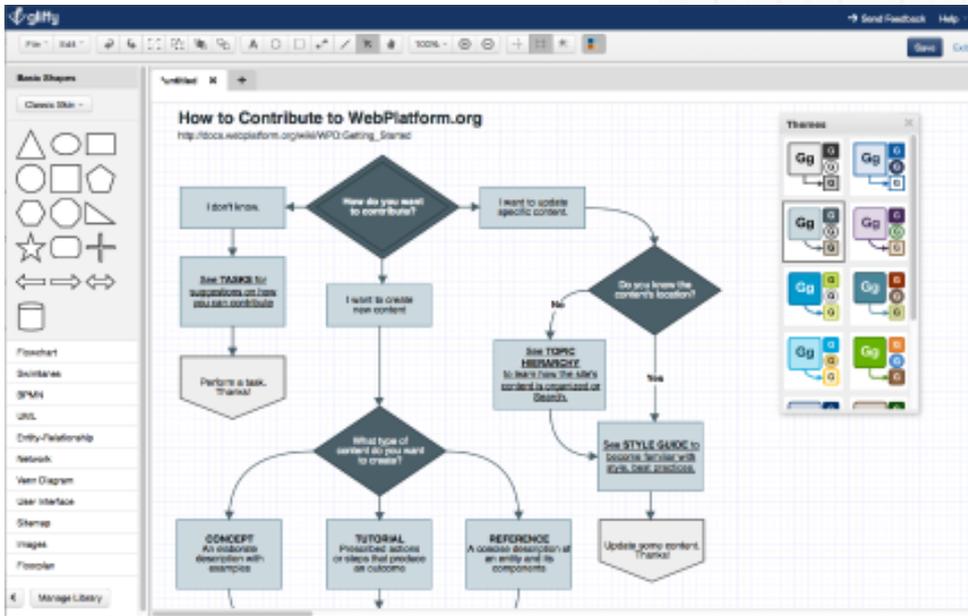
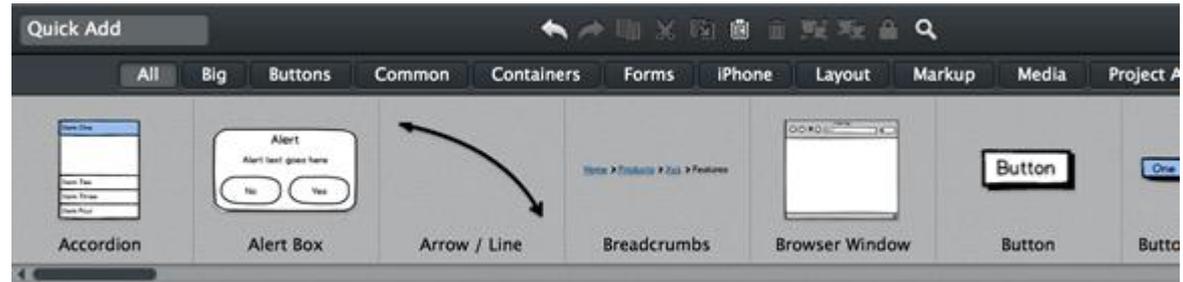
10 days remaining

Epics

Large Scale

Jira

Balsamiq Mockups



Gliffy (Diagramme, UML)





Fragen ?

Orientation in Objects GmbH

Weinheimer Str. 68
68309 Mannheim

www.oio.de
info@oio.de



Vielen Dank für Ihre Aufmerksamkeit !

Orientation in Objects GmbH

Weinheimer Str. 68
68309 Mannheim

www.oio.de
info@oio.de