

Dokumentation am (Riesen-)Beispiel

arc42, AsciiDoc und Co. in Aktion

Ralf D. Müller & Gernot Starke



Dr. Gernot Starke

innoQ Fellow



Softwarearchitektur

- Entwurf
- Evolution + Modernisierung
- Dokumentation
- Reviews



+49 177 7282570

gernot.starke@innoq.com



Ralf D. Müller

Deutsche Bank

DIGITAL
FACTORY



Bei Tag Solution Architect
in der Digital Factory
der Deutschen Bank.

In der Freizeit Geek:

- Web-Technologien
- Qualität (Security, Testautomation)
- Produktivität (Gradle, Groovy, Grails)

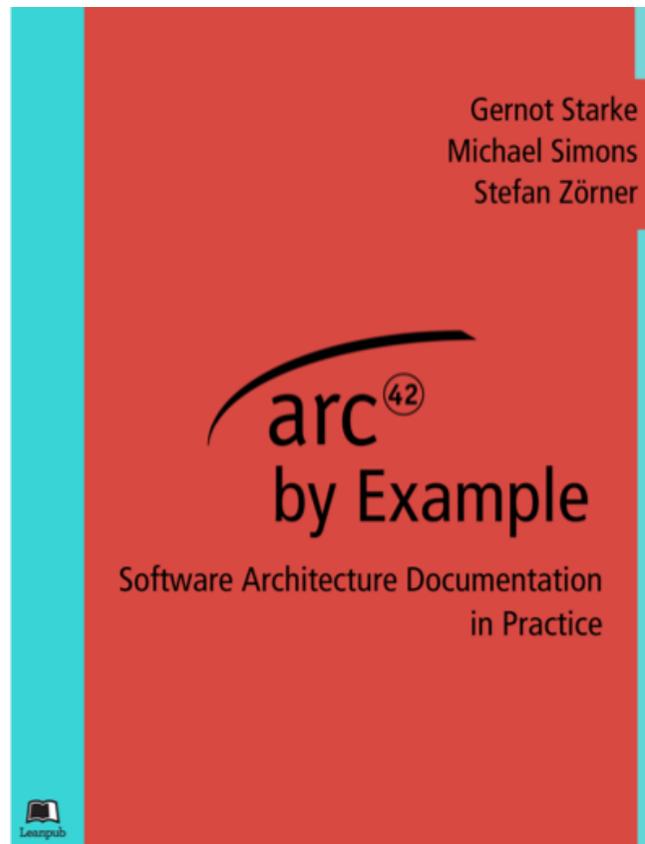
Maintainer von docToolchain

The image shows a Twitter profile for **docToolchain** (@docToolchain) with 225 tweets, 38 followers, and 73 people following. Below the profile is a complex dependency diagram for the docToolchain project. The diagram is a flowchart showing the relationships between various components. A legend indicates that yellow boxes represent 'your documentation', green boxes represent 'output document', blue boxes represent 'intermediate document', white boxes represent 'build task', red boxes represent 'software component', and grey boxes represent 'proposed'. Solid lines indicate the 'direction of control', while dashed lines indicate 'proposed' relationships. The diagram shows a central 'your documentation' component that interacts with 'Diagram Plugins' and 'Element Rules'. These components then interact with 'Diagram Plugins' and 'Diagram Plugins' which in turn interact with 'Diagram Plugins' and 'Diagram Plugins'. The diagram also shows interactions with 'Diagram Plugins' and 'Diagram Plugins' which interact with 'Diagram Plugins' and 'Diagram Plugins'. The diagram is a complex network of dependencies and control flows.



42% off

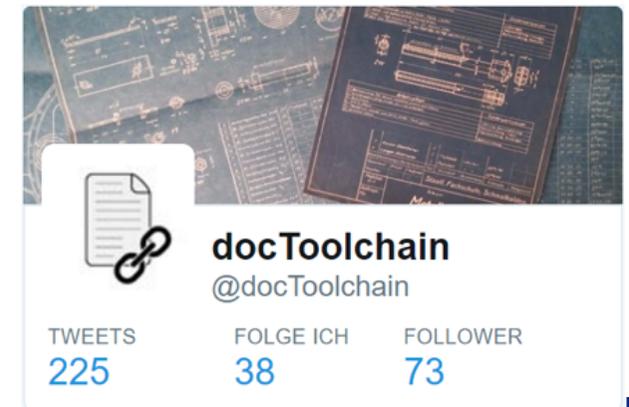
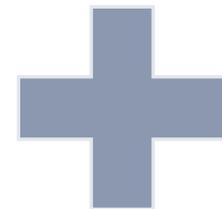
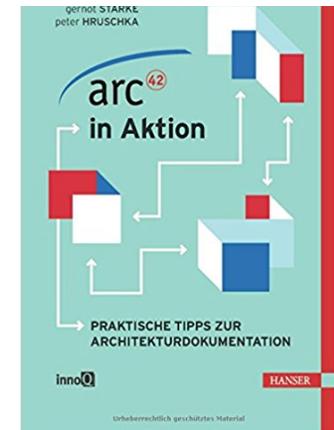
<https://leanpub.com/arc42byexample/c/JAX2017>
(gültig bis 10. Juni 2017)



Das erwartet Sie !

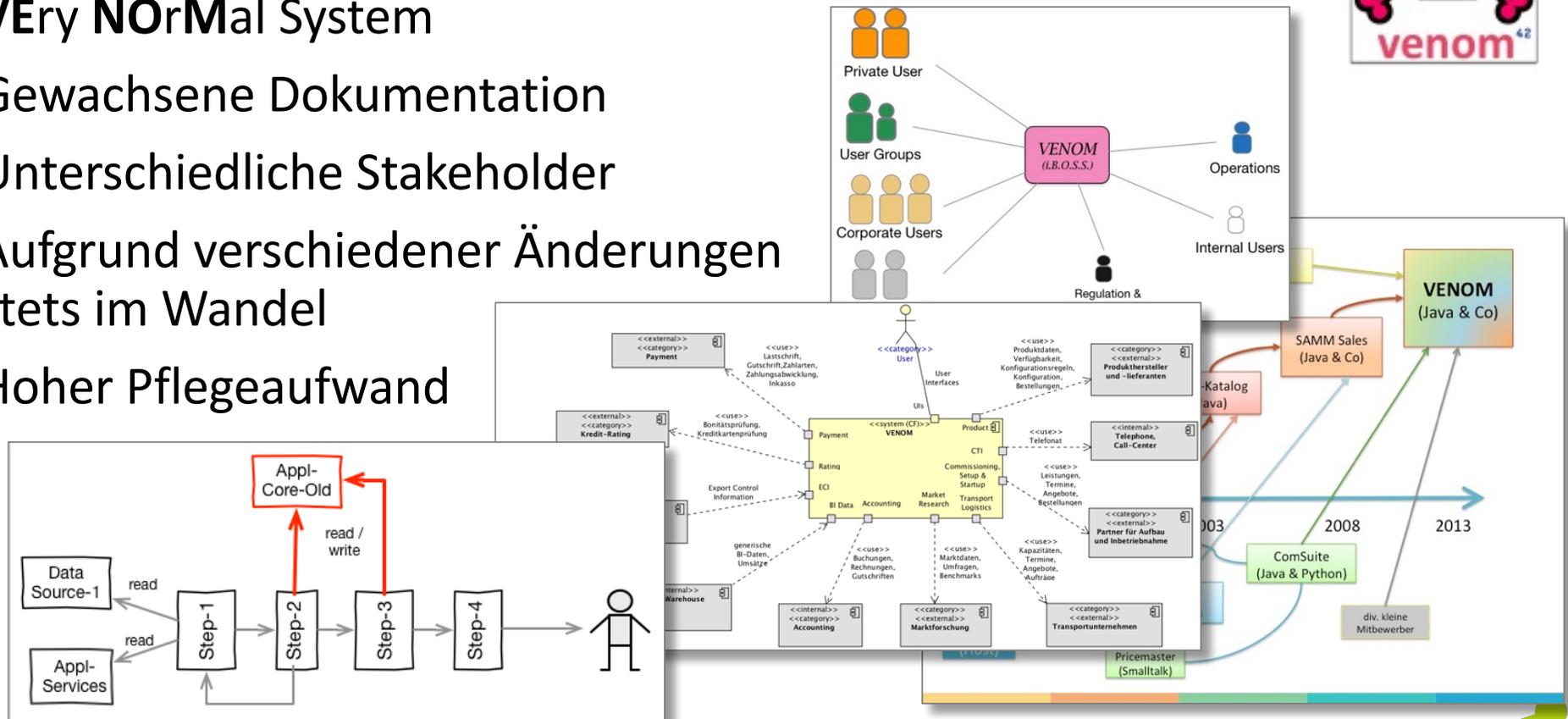
- praktische **Architekturdokumentation**
- Tipps zu **arc42** und **docs-like-code**
- Experimentelle Features :-)
- Vorschläge aus Erfahrung

NSB (No Silver Bullet)



Das VENOM Projekt

- **VERY NORMAL** System
- Gewachsene Dokumentation
- Unterschiedliche Stakeholder
- Aufgrund verschiedener Änderungen stets im Wandel
- Hoher Pflegeaufwand



Darf ich vorstellen? Geoff – Solution Architect

- Solution Architect für VENOM
 - solider technischer Background

Aufgaben (u.a.):

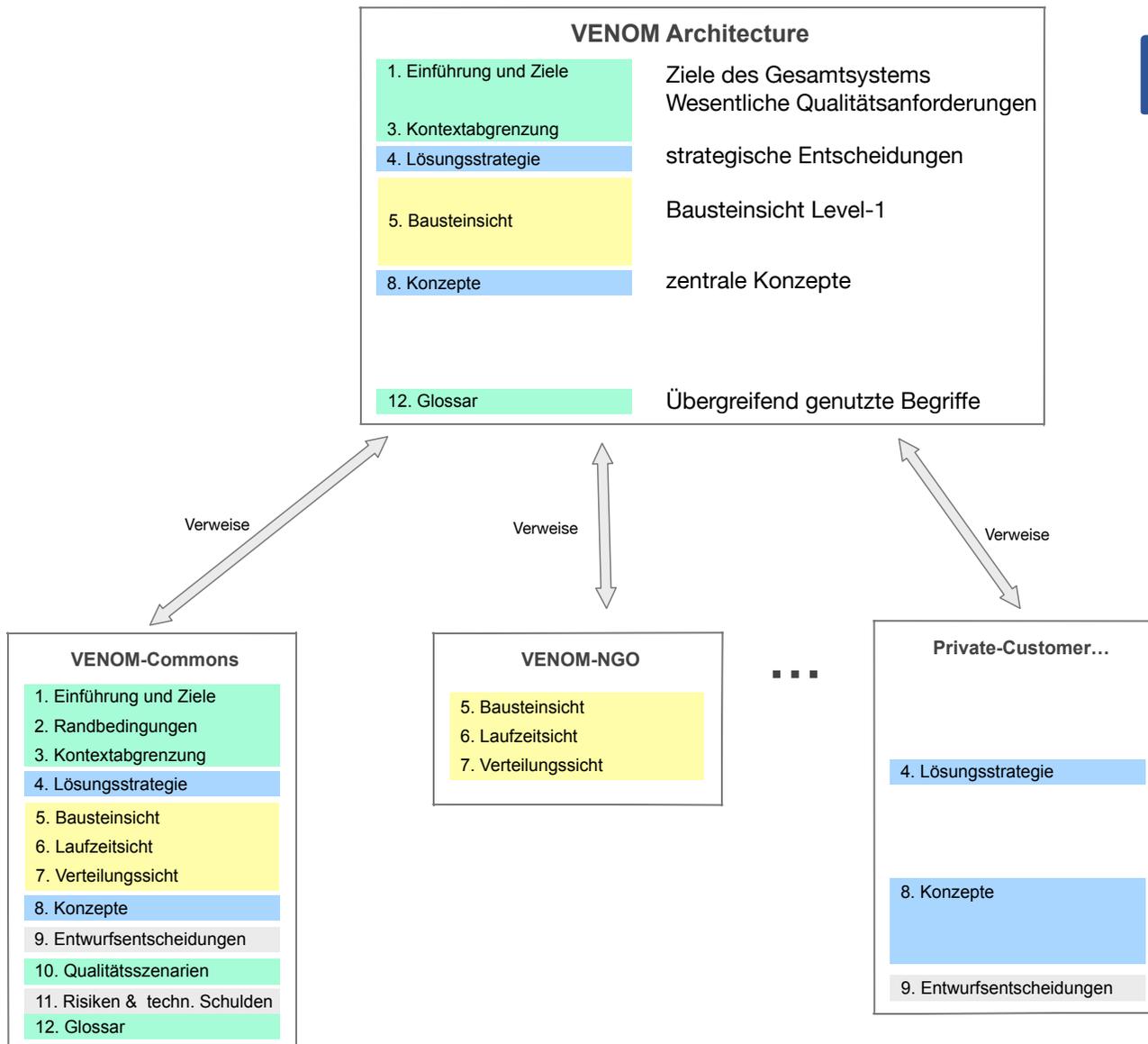
- Evolution des Systems
- Dokumentation + Kommunikation der Architektur



Methodik-

Modularisiere Dokumentation großer Systeme!

Eigene Dokumente für relevante Teilsysteme



Siehe: <http://faq.arc42.org/questions/l-1/>



Trenne Projekt- von Systemdokumentation!

Methodik-



Team



Projektdokumentation

~arc42 „locker“

Diskussionen

Implementation-Guide,

Tasks / Issues

„Gärtner“



Systemdokumentation

1. Einführung und Ziele
2. Randbedingungen
3. Kontextabgrenzung
4. Lösungsstrategie
5. Bausteinsicht
6. Laufzeitsicht
7. Verteilungssicht
8. Konzepte
9. Entwurfsentscheidungen
10. Qualitätsszenarien
11. Risiken
12. Glossar

Weitere relevante Infos...

(Betrieb/Admin, Test, Release...)



- [Tip 3-5: Restrict the context to an overview, avoid too many details!](#)
- [Tip 3-6: Simplify the context by categorization!](#)
- [Tip 4-1: Explain the solution strategy as compact as possible \(e.g. as list of keywords\)!](#)
- [Tip 4-2: Describe the solution approaches as a table!](#)
- [Tip 4-5: Let the solution strategy grow iteratively / incrementally!](#)
- [Tip 5-3: Always describe level-1 of the building block view \('Level-1 is your friend'\)!](#)
- [Tip 5-6: Hide the inner workings of blackboxes!](#)
- [Tip 6-2: Document only a few runtime scenarios!](#)
- [Tip 6-3: Document 'schematic' \(instead of detailed\) scenarios!](#)
- [Tip 6-9: Use a textual notation to describe runtime scenarios!](#)
- [Tip 7-7: Use tables to document software/hardware mapping!!](#)
- [Tip 8-1: Explain the Concepts!](#)
- [Tip 8-9: Document decisions instead of concepts!](#)
- [Tip 9-1: Document only architecturally relevant decisions!](#)
- [Tip 9-7: Document decisions informally as a blog \(RSS-feed\)!](#)
- [Tip 12-2: Document the glossary as a table!](#)
- [Tip 12-5: Keep the glossary compact! Avoid trivia.](#)
- [Tip 12-6: Make your 'product owner' or 'project manager' responsible for the glossary](#)
- [Tip 1-16: Describe only the top 3-5 quality goals in the introduction!](#)

- [Tip 1-22: Skip the stakeholder table if your management already maintains it!](#)
- [Tip 1-23: Classify your stakeholders by interest and influence!](#)
- [Tip 3-17: Combine business context with technical information!](#)
- [Tip 3-19: Defer technical context to the deployment view!](#)
- [Tip 5-10: Use crosscutting concepts to describe or specify similarities in building blocks!](#)
- [Tip 5-15: Align the mapping of source-code to building-blocks along the directory and file structure!](#)
- [Tip 6-10: Use both small and large building blocks in scenarios!](#)
- [Tip 7-10: Leave hardware decisions to hardware-experts!](#)
- [Tip 8-10: Use the collection from arc42 as checklist for concepts!](#)
- [Tip 5-21: Describe or specify internal interfaces with minimal effort!](#)

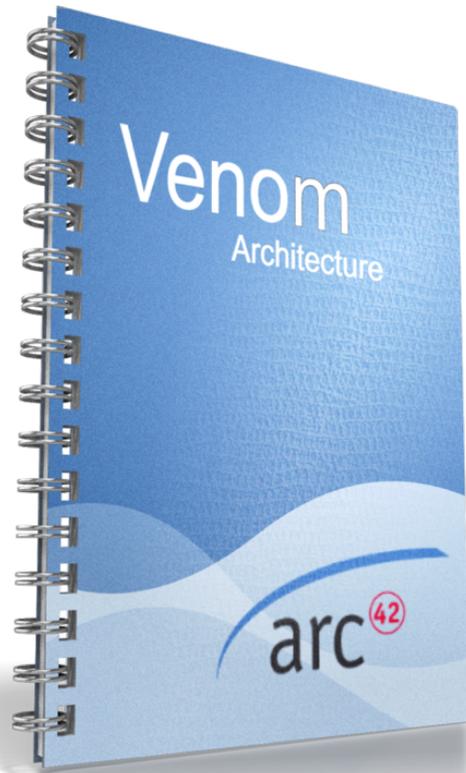
Methodik-

Dokumentiere sparsam!

Siehe: <http://docs.arc42.org/keywords/#lean>



Format der Dokumentation



- MS Word: etablierter Standard
- arc42 in vielen Formaten:
 - docx
 - asciidoc ←
 - markdown
 - latex
 - html
 - textile
 - confluence





arc42 Formate

- AsciiDoc aus unserer Sicht flexibelstes Format
- In alle anderen Formate (fast verlustfrei) transformierbar, daher immer „Plan B“



Demo – eine erste Konvertierung

build.gradle	demo.adoc	console	output
<pre>plugins { id "org.asciidoctor.convert" version "1.5.3" }</pre>			



Demo – eine erste Konvertierung

build.gradle

demo.adoc

console

output

```
= A first Headline
```

```
And a first paragraph.
```

```
It continuous on the next headline
```

```
Second paragraph.
```

```
== Second-Level Headline
```

```
A link to http://asciidoctor.org/docs[Asciidoctor.org]
```



Demo – eine erste Konvertierung

build.gradle	demo.adoc	console	output
<pre>PS C:\Users\Demo\jax2017\demo1> gradle asciidoc :asciidoctor io/console not supported; tty will not be manipulated BUILD SUCCESSFUL Total time: 4.554 secs PS C:\Users\Demo\jax2017\demo1></pre>			



Demo – eine erste Konvertierung

build.gradle	demo.adoc	console	output
<p>A first Headline</p> <p>And a first paragraph. It continous on the next headline</p> <p>Second paragraph.</p> <p>Second-Level Headline</p> <p>A link to Asciidoctor.org</p> <p>Last updated 2017-04-21 18:01:22 +02:00</p>			

<http://asciidoctor.org/docs/render-documents/>





Tools zur Konvertierung

- Geringste Einstiegshürde:
Gradle und asciidoctorj
- Maven ist aufwändiger, gut unterstützt
- Gradle bezüglich weiterer Build-Steps flexibler



Out-of-the-Box: docs-as-code

- „ablenkungsfrei“ –
Dokumentation wie Code oder eMails schreiben
 - modularisierbar
 - Bilder referenziert, nicht eingebettet
 - Integration von Source-Code
-
- Reviews, Pull-Requests, Versionierung durch Git
 - Konvertierung nach HTML5, DocBook u.v.a.





Unterdokumente

- Setzen von `imagedir` am Anfang jedes Dokuments:

```
ifndef::imagesdir[:imagesdir: ../../images]
```

- Partielle Includes

```
include::subdocument.adoc[tags=xyz]
```

- Korrektur von Überschriften-Level

```
include::subdocument.adoc[tags=xyz, leveloffset=+1]
```



...die Reise beginnt erst

- Geoff bisher zufrieden
- alte Dokumentation überführen...

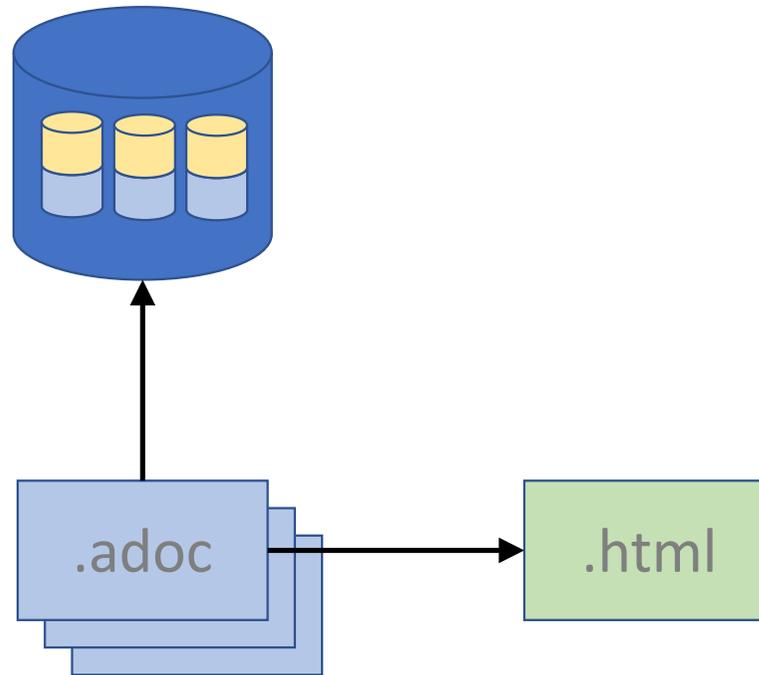


treat Docs-as-Code

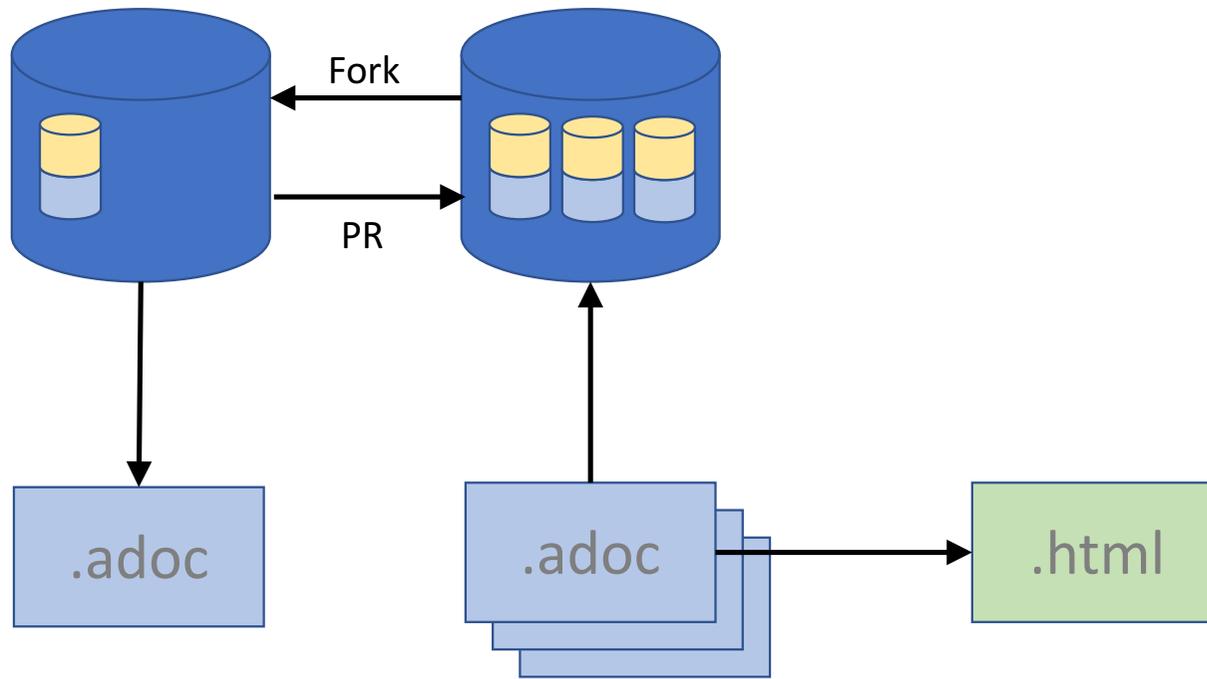
- Geoff erkennt, dass die Transformation nach AsciiDoc erst der Anfang war
- Als nächstes möchte er durch den Docs-as-Code Ansatz die Überarbeitung der Dokumentation weiter vereinfachen



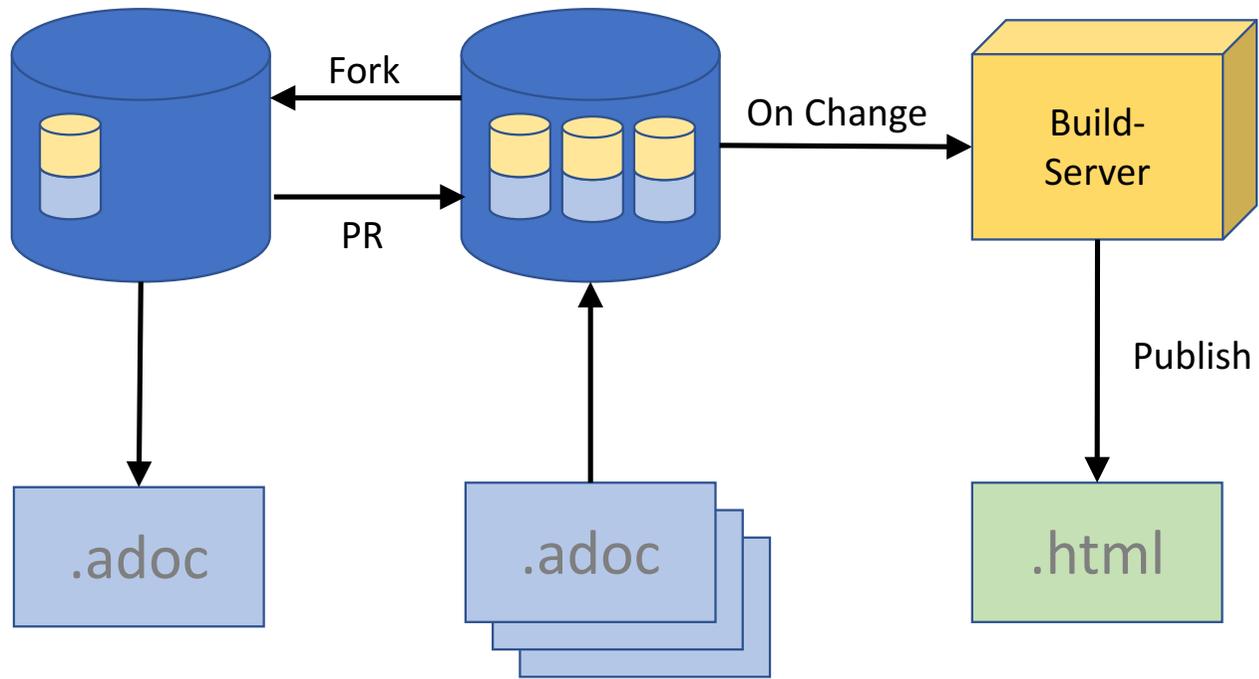
treat Docs-as-Code I: Version Control



treat Docs-as-Code II: Git-Flow



treat Docs-as-Code III: Build-Server



Diagramme

- Geoff stört der hohe Pflegeaufwand für Diagramme
- Beherrscht AsciiDoc nicht PlantUML?



Diagramme: PlantUML

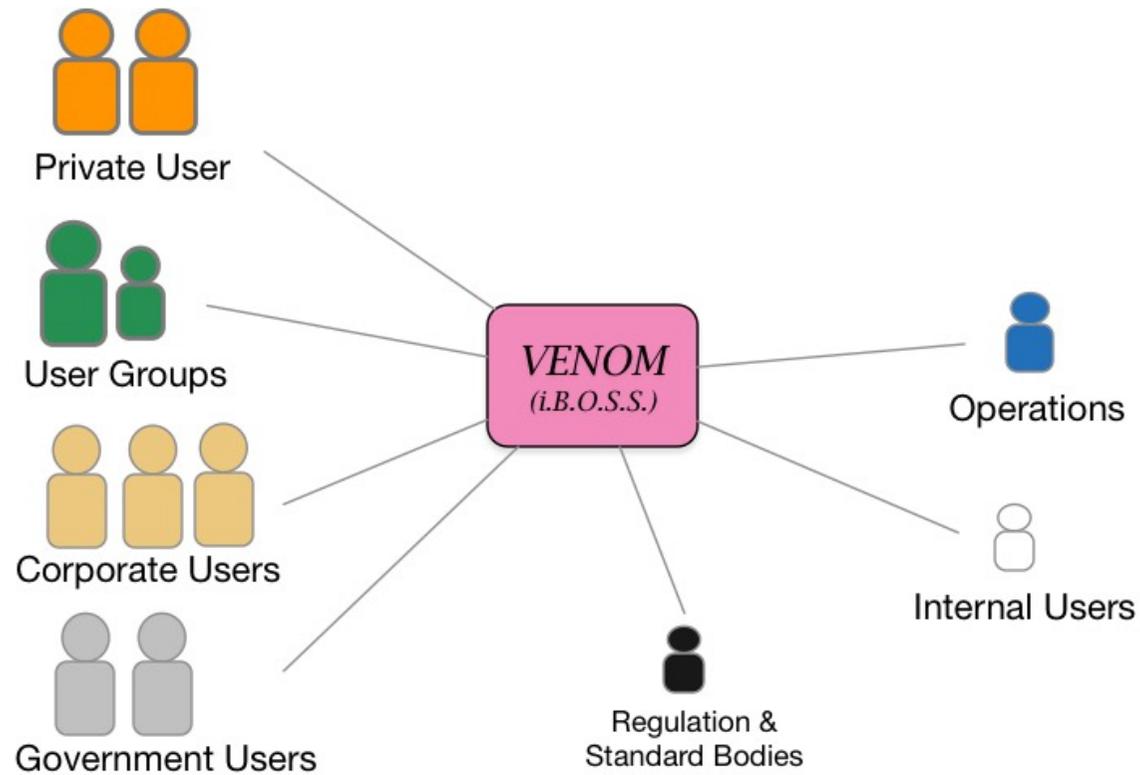


Diagramme: PlantUML

```
.Benutzer und Benutzergruppen von VENOM
```

```
[plantuml
```

```
----
```

```
!pragma graphviz_dot jdot
```

```
:Private User: as private
```

```
:User Groups: as groups
```

```
:Corporate Users: as corporate
```

```
:Government Users: as gov
```

```
:Regulation &\nStandard Bodies: as bodies
```

```
:Operations: as ops
```

```
:internal Users: as internal
```

```
(VENOM\ni.B.O.S.S) as venom
```

```
private -right-> venom
```

```
groups --> venom
```

```
corporate --> venom
```

```
gov -up-> venom
```

```
bodies -up-> venom
```

```
ops --> venom
```

```
internal -left-> venom
```

```
----
```

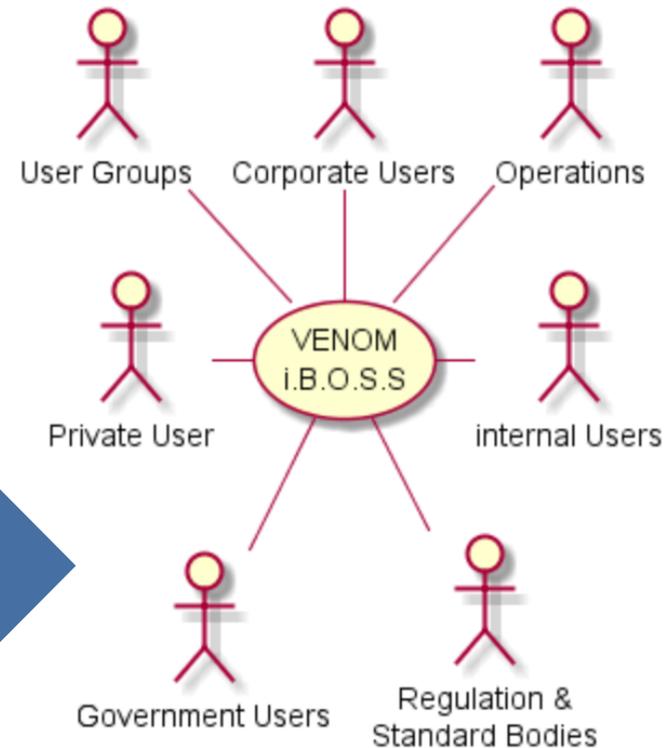
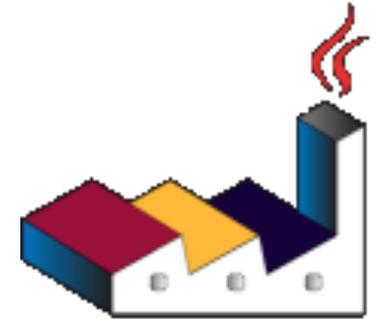


Abbildung 2. Benutzer und Benutzergruppen von VENOM

Diagramme als Plain-Text: PlantUML

- <http://plantuml.com/>
- <http://asciidoctor.org/docs/asciidoctor-diagram/>



Nicht alle Diagrammtypen sind für PlantUML gleichgut geeignet. Sequenzdiagramme sind jedoch ein sehr guter Anwendungsfall!





Asciidoctor Plugins

- AsciidoctorJ oder jRuby?
- Bei PlantUML Verschiedene Output-Formate beachten!





Diagramme

- Im Zweifel Pfeile immer vom Aufrufenden zum Aufgerufenen
- Noch keinen eigenen Stil gefunden?
=> C4 von Simon Brown ist ein guter Start
http://www.codingthearchitecture.com/2014/08/24/c4_model_poster.html

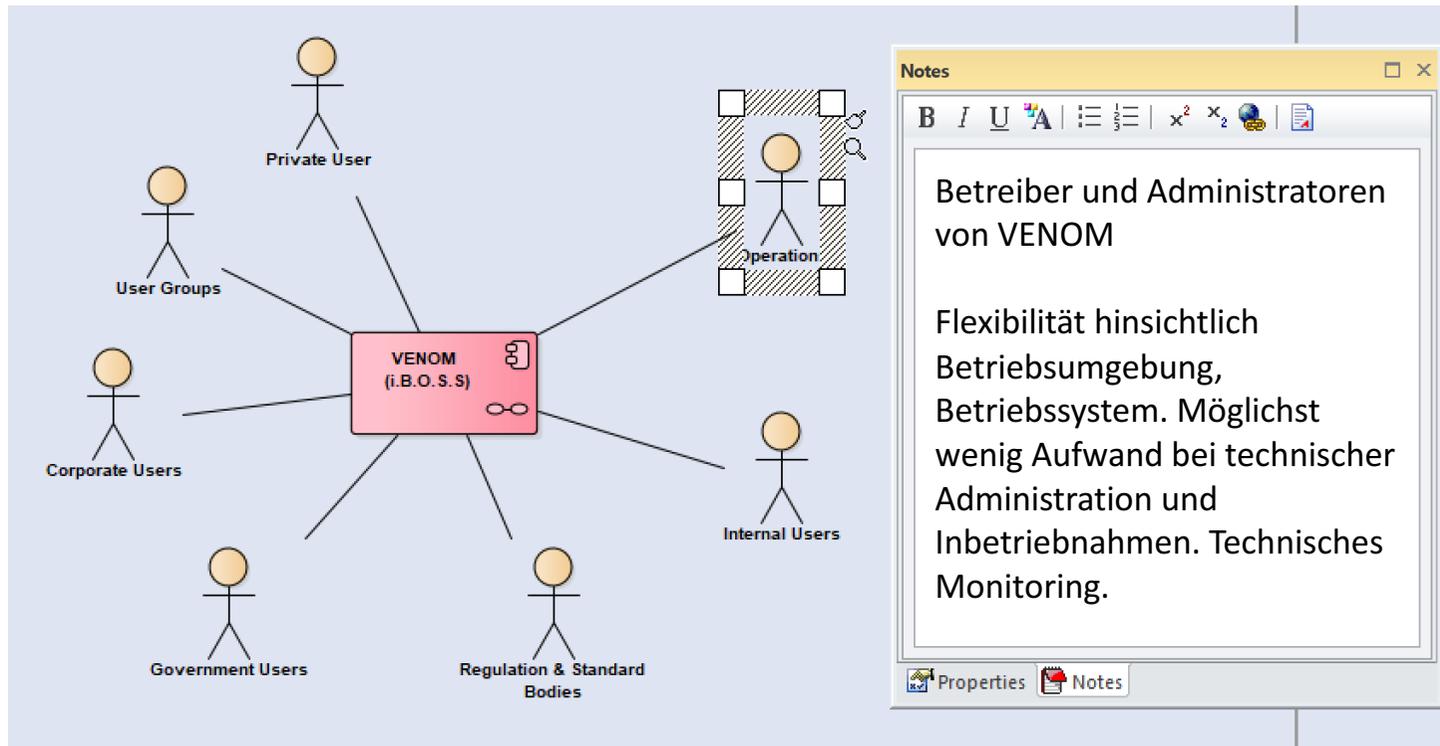


Diagramme: Modellierung

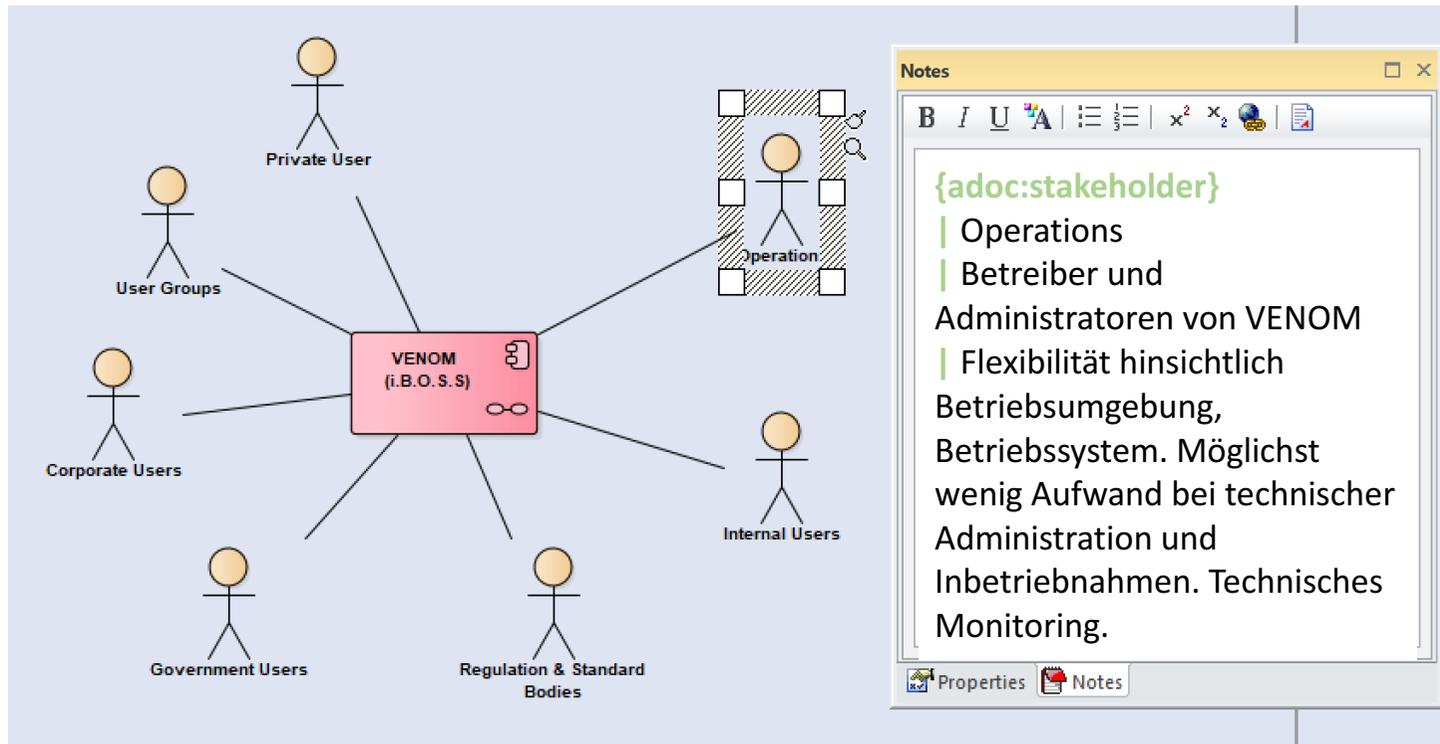
- Geoff nutzt ein UML-Modellierungstool
- Einbetten der Grafiken schwerfällig
- Notizen im UML-Modell gehen in Doku verloren



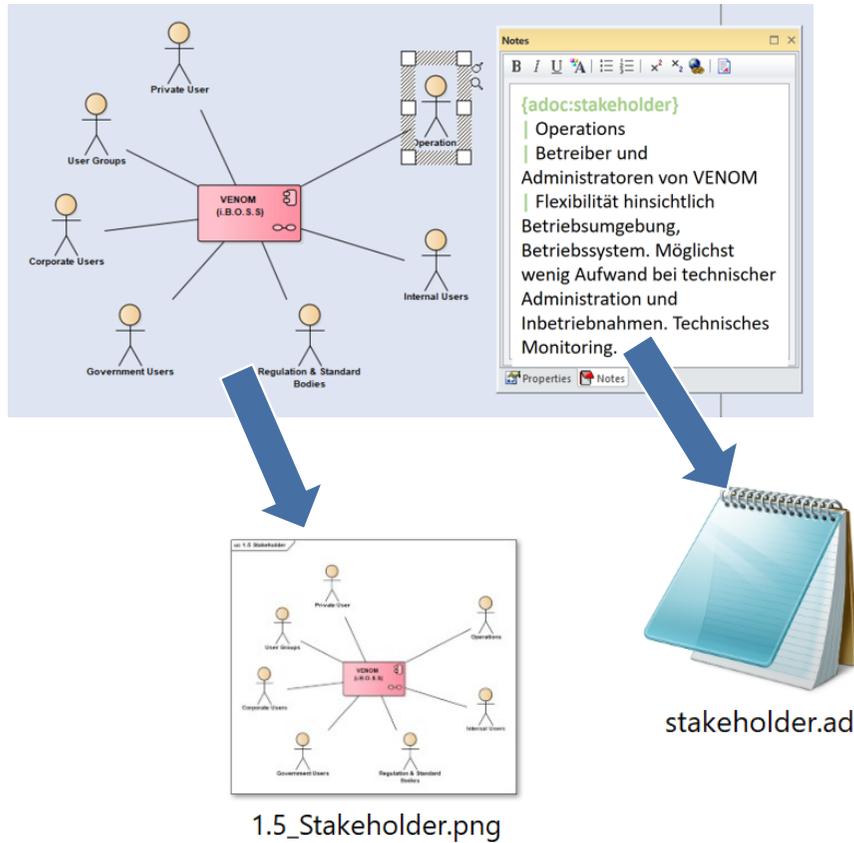
treat Docs-as-Code IV: automate



treat Docs-as-Code IV: automate



treat Docs-as-Code IV: automate



=== Stakeholder

==== Benutzer und Benutzergruppen

[[figure-users]]

image::ea/1.5_Stakeholder.png[title="Benutzer und Benutzergruppen von VENOM"]

[cols="2,3,3,2" options="header"]

.Benutzer und Benutzergruppen

|===

| Rolle | Beschreibung | Ziel | Bemerkungen

include::.../ea/stakeholder.ad[]

|===

treat Docs-as-Code IV: automate

VENOM: Architecture Documentation 

1.5. Stakeholder

1.5.1. Benutzer und Benutzergruppen

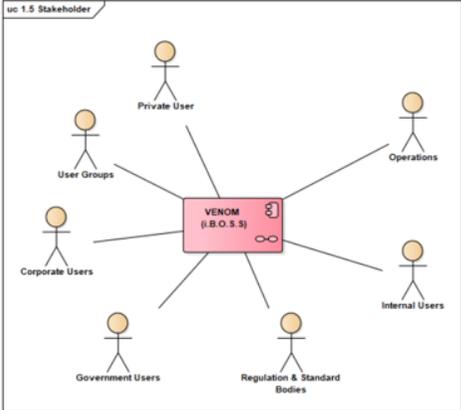


Abbildung 2. Benutzer und Benutzergruppen von VENOM

Tabelle 3. Benutzer und Benutzergruppen

Rolle	Beschreibung	Ziel	Bemerkungen
Corporate User	(CUs) Unternehmen oder kommerzielle Organisationen	Möchten komplexe Produkte fehlerfrei konfigurieren und mit Aufbau und Inbetriebnahme wenig Aufwand haben.	

| 9



Stakeholder

- Geoff bemerkt schnell, dass nicht jeder mit einer online HTML-Dokumentation glücklich ist
- Er muss für unterschiedliche Stakeholder die Dokumente auch unterschiedlich aufbereiten



.docx bzw. MS Word

<http://pandoc.org>

Pandoc a universal document converter

[Donate](#) [Flattr](#) 0

- [About](#)
- [Installing](#)
- [Getting started](#)
- [Demos](#) ▾
- [Documentation](#) ▾
- [Help](#)
- [Extras](#)
- [Releases](#)

About pandoc

If you need to convert files from one markup format into another, pandoc is your swiss-army knife. Pandoc can convert documents in [markdown](#), [reStructuredText](#), [textile](#), [HTML](#), [DocBook](#), [LaTeX](#), [MediaWiki markup](#), [TWiki markup](#), [OPML](#), [Emacs Org-Mode](#), [Txt2Tags](#), Microsoft Word [docx](#), LibreOffice [ODT](#), [EPUB](#), or [Haddock markup](#) to

- HTML formats: XHTML, HTML5, and HTML slide shows using [Slidy](#), [reveal.js](#), [Slideous](#), [S5](#), or [DZSlides](#).
- Word processor formats: Microsoft Word [docx](#), OpenOffice/LibreOffice [ODT](#), [OpenDocument XML](#)
- Ebooks: [EPUB](#) version 2 or 3, [FictionBook2](#)
- Documentation formats: [DocBook](#), [TEI Simple](#), [GNU TexInfo](#), [Groff man](#) pages, [Haddock markup](#)
- Page layout formats: [InDesign ICML](#)
- Outline formats: [OPML](#)
- TeX formats: [LaTeX](#), [ConTeXt](#), LaTeX Beamer slides
- [PDF](#) via LaTeX
- Lightweight markup formats: [Markdown](#) (including [CommonMark](#)),



...bzw. pdf



3. Kontextabgrenzung

Die folgenden Unterkapitel zeigen die Einbettung unseres Systems in seine Umgebung.

3.1. Fachlicher Kontext

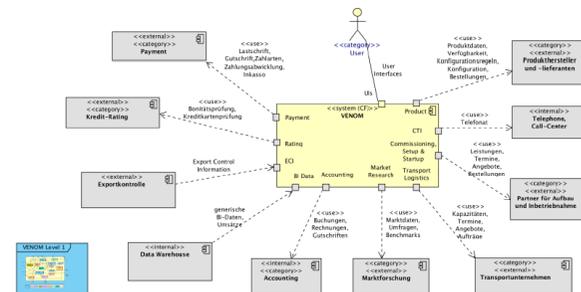


Abbildung 3. Business Context

Der Stereotyp «Category» bezeichnet Schnittstellen, zu denen es unterschiedliche Ausprägungen, Varianten oder Anbieter gibt.

Tabelle 5. Fachlicher Kontext

Nachbar	Beschreibung
ProduktHersteller	Küchen-, Möbel-, Lagerystemhersteller. Stellen Produktdaten und Verfügbarkeitsdaten bereit, teilweise Konfigurationsdienste oder -komponenten, die in VENOM integriert werden.
externe Partner für Aufbau u. Inbetriebnahme	Handwerker, Montreure, Ingenieurbüros, teilweise Gutachter. Insbesondere werden hier Termine und Verfügbarkeiten abgestimmt. Allgemeine organisatorische und kaufmännische Fragen der Zusammenarbeit werden im Vorfeld vertraglich geregelt bzw. über die Partner-Schnittstelle erfragt.
Telephone, Call-Center	(internes) Call-Center für Post-Sales und Support-Aufgaben. Behandeln Rückrufwünsche und Anfragen zu Produkten sowie Unterstützung gerade online laufender Konfigurationen. Hier arbeiten einige der hervorragenden Experten von SAMM, um Kunden bei hochgradig komplexen Produkten zu unterstützen.
Transportunternehmen	Angebotsanfragen, Buchung und Beauftragung von Transportkapazitäten. In der Regel arbeitet VENOM mit einigen Speditionen zusammen, in Ausnahmefällen werden Transportunternehmer direkt beauftragt. Für Kleinbestellungen werden direkt entsprechende Versender (Post, etc.) beauftragt.
Data Warehouse	Angebote, Aufträge, Marketingkampagnen zur Auswertung und Entscheidungsunterstützung (Business Intelligence).
Accounting	Buchhaltung, unter anderem SAP. Relevante Buchungen werden in das jeweilige Buchhaltungssystem geschrieben.
Marktforschung	Konkurrenzanalyse, Auswertung von Web-Analytics, Preisbeobachtung
Exportkontrolle	Manche Produkte (etwa: Komplexe pharmazietaugliche Transportsysteme) können in einigen Ländern/Regionen Ausfuhrrestriktionen unterliegen. Hierüber können die Behörden entsprechende Anfragen stellen bzw. erhalten dazu Resultate. In Deutschland ist das das Bundesamt für Wirtschaft und Außenkontrolle, BAFÄ.
(Kredit-) Rating	Bonitätsprüfung und Risikobewertung (Rating oder Scoring) von Kunden, beispielsweise durch Kreditkartenprüfung oder Bonitätsauskunft. Je nach Kundenart (Privat-, Firmenkunde, Organisationen) werden hier verschiedene Prüfungen durchgeführt. Verschiedene Anbieter, je nach Land. In Deutschland beispielsweise Arvato-InfoScore oder Schufa Holding. Diese Schnittstellen sind durch die jeweiligen Anbieter vorgegeben.
Payment	Externe Dienstleistungen in Bezug auf Zahlungen und Zahlungsbuchung. Unterschiedliche Anbieter, unter anderem Arvato-InfoScore, PayPal und PayMill, sowie SEPA-Schnittstellen zu diversen Banken. In kritischen Fällen auch Beauftragung von Inkassodienstleistungen.

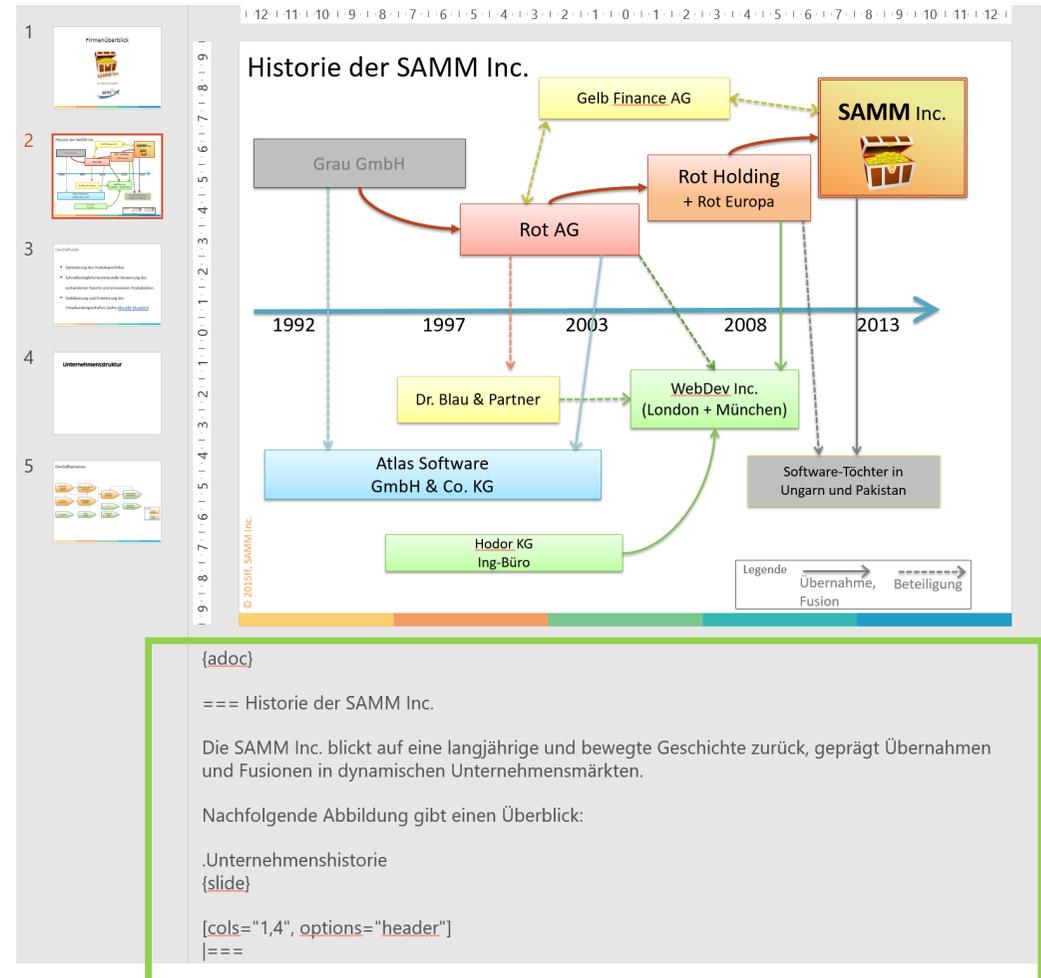


Export PPT

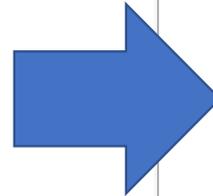
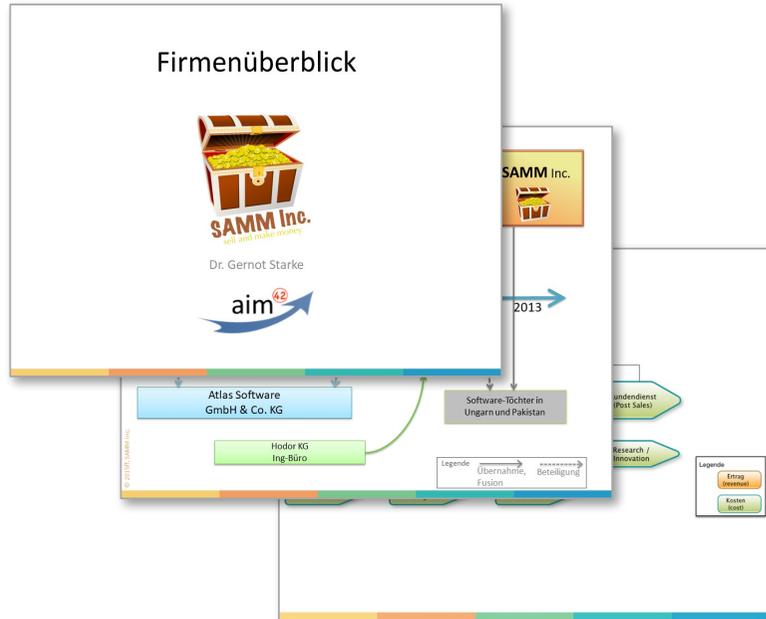
Aufgabe:

Powerpoint in Doku integrieren:

- Sprechernotizen enthalten asciidoc
- Slides und asciidoc werden automatisch exportiert



Bonus: Export PPT



Inhaltsverzeichnis

- 1. Firmenüberblick 2
 - 1.1. Historie der S.A.M.M. Inc. 2
 - 1.2. Geschäftsziele..... 3
- 2. Unternehmensstruktur 4
 - 2.1. Geschäftsprozesse 4
- Anhang A: Image Credits 5
- Anhang B: Lizenz..... 5

Geschichte zurück, Unternehmensmärkten.

2013

Software-Töchter in Ungarn und Pakistan

Übernahme, Fusion, Beteiligung

© seit 2003 S.A.M.M. Inc.

Legende: —> Übernahme, —> Fusion, —> Beteiligung

Ertrag (Revenue) / Kosten (Cost)

die Geschäftsprozesse bei

Legende: —> Ertrag (Revenue), —> Kosten (Cost)

© seit 2003 das strategische Wachstum der Segment unterstützen. Hat mittlerweile ca. 28% Anteile an Gold Finance hält.

rig in Deutschland und Europa. Nach dem, was stark erweitert und in Europa kleinere Standorte und steigere dadurch Ertrag.

eine Aktiengesellschaft wirtschaftlich und

gültige oder ausführeschlechte Märkte.

ändern für komplexe Produkte wie (Industrieanlagen). Frühzeitige Investition in

WebDev. Inc.	Geprägt als Anbieter von e-commerce Dienstleistungen, insbesondere Softwareentwicklung. Langjährige Erfahrung in webbasierten Verkaufsplattformen, Konzeption und Entwicklung von Webshops, Integration von online-Zahlungsdiensten. 2002 erwirbt die Net AG vom Anstalt, 2010 übernimmt S.A.M.M. Inc. die WebDev zu 100%.
--------------	--

2 |

4 |

Confluence

- Aber alle anderen Dokumente sind in Confluence...
- Confluence speichert die Seiten intern als xhtml
- ...und hat eine REST-API
- et voilà...



Zusammenarbeit

The screenshot shows a web application interface with a sidebar on the left and a main content area on the right. The sidebar contains navigation elements like 'Bereiche', 'Personen', and a list of sections under 'SEITENHIERARCHIE'. The main content area displays the title '1.2. Das VENOM-System' and a sub-section '1.2.1. Historie'. Below the text is a timeline diagram showing the evolution of the VENOM system from 1992 to 2013, with various predecessor systems like 'WaWi', 'Backoffice-Katalog', 'Web-Katalog', 'Samm Sales', 'eGov', 'ComSuite', and 'Campaigner' leading to the final 'VENOM (Java & Co)' system.

Seiten / 1. Einführung Bearbeiten Favorit Beobachten Teilen ...

1.2. Das VENOM-System

Erstellt von docToolchain gestern um 10:59 PM

- 1.2.1. Historie

1.2.1. Historie

VENOM wurde über die letzten Jahre durch die SAMM Inc. und ihre Vorgängerunternehmen entwickelt. Das System in seiner heutigen Form ist aus einer Vielzahl von Vorgängersystemen entstanden, erklärbar im Wesentlichen durch die Unternehmenshistorie (siehe dazu die SAMM Inc. Unternehmensinformation)

Einen groben Überblick über die Historie von VENOM und dessen Vorgänger gibt Abbildung [Historie von VENOM](#).

```
graph TD; WaWi["„WaWi“ (Host, Cobol)"] --> Backoffice["Backoffice-Katalog (Java + Host)"]; Backoffice --> WebKatalog["Web-Katalog (Java)"]; WebKatalog --> SammSales["Samm Sales (Java & Co)"]; eGov["eGov (Python)"] --> SammSales; ComSuite["ComSuite (Java & Python)"] --> SammSales; Campaigner["Campaigner (Java, PHP)"] --> VENOM["VENOM (Java & Co)"]; SammSales --> VENOM;
```



Zusammenarbeit

The screenshot shows a web interface for a Wiki page. On the left is a navigation sidebar with sections like 'BEREICHsverknüpfungen', 'SEITENhierarchie', and '1. Einführung'. The main content area displays a diagram with nodes: '„Atlas 1“ (Host)', '„Atlas 2“ (AS/400, Cobol)', 'Pricemaster (Smalltalk)', and 'div. kleine Mitbewerber'. Below the diagram is the title 'Abbildung 1. Historie von VENOM' and a paragraph of text. A bulleted list follows, describing various issues. At the bottom, a red box highlights the 'Gefällt mir' button, the text 'Sei der Erste, dem dies gefällt.', and a text input field for comments.

Abbildung 1. Historie von VENOM

Aufgrund der langen Historie und der häufig wechselnden Entwicklungs- und Wartungsteams sowie der damit verbundenen wechselnden Verantwortlichkeiten und Prioritäten leidet VENOM unter folgenden grundsätzlichen Problemen (Details werden an anderer Stelle beschrieben):

- Signifikante Steigerung von Entwicklungs- und Betriebskosten
 - Hohe Aufwände zur Entwicklung und Inbetriebnahme neuer Features. Teilweise dauert es auch bei wichtigen Anforderungen mehr als 4-6 Monate, bis die entsprechende Funktion in Betrieb gehen kann.
 - Hohe Fehleranfälligkeit bei Änderungen und Neuentwicklungen: Selbst kleine Änderungen ziehen substantielle Test- und Bugfixing-Aufwände nach sich.
- Sinkende Betriebsstabilität: Die Anzahl an Fehlern im laufenden Betrieb (Prio-1 Produktionsfehler) nimmt zu.
- Die Verlässlichkeit von Aufwandsschätzungen der Entwicklungsteams nimmt ab.
- Mangelhafte Qualität von Dokumentation

Gefällt mir Sei der Erste, dem dies gefällt. Keine Stichwörter

Schreiben Sie einen Kommentar



Zusammenarbeit

arc Bereiche Personen Erstellen

Bereichsverknüpfungen
Hier können Sie Verknüpfungen zu Ihren wichtigsten Inhalten vornehmen.

Seiten / 1. Einführung Bearbeiten Favorit Beobachten Teilen

„Atlas 1“ (Host) „Atlas 2“ (AS/400, Cobol) Pricemaster (Smalltalk) div. kleine Mitbewerber

Abbildung 1. Historie von VENOM

Aufgrund der langen Historie und der häufig wechselnden Entwicklungs- und Wartungsteams sowie der damit verbundenen wechselnden Verantwortlichkeiten und Prioritäten leidet VENOM unter folgenden grundsätzlichen Problemen (Details werden an anderer Stelle beschrieben):

- Signifikante Steigerung von Entwicklungs- und Betriebskosten
 - **Hohe Aufwände zur Entwicklung** und Inbetriebnahme neuer Features. Teilweise dauert es auch bei wichtigen Anforderungen mehr als 4-6 Monate, bis die entsprechende Funktion in Betrieb gehen kann.
 - Hohe Fehleranfälligkeit bei Änderungen und Neuentwicklungen: Selbst kleine Änderungen ziehen substantielle Test- und Bugfixing-Aufwände nach sich.
- Sinkende Betriebsstabilität: Die Anzahl an Fehlern im laufenden Betrieb (Prio-1 Produktionsfehler) nimmt zu.
- Die Verlässlichkeit von Aufwandsschätzungen der Entwicklungsteams nimmt ab.
- Mangelhafte Qualität von Dokumentation

Gefällt mir Sei der Erste, dem dies gefällt. Keine Stichwörter

Schreiben Sie einen Kommentar...

Bereich konfigurieren



Zusammenarbeit

Abbildung 1. Historie von VENOM

Aufgrund der langen Historie und der häufig wechselnden Entwicklungs- und Wartungsteams sowie der damit verbundenen Verantwortlichkeiten und Prioritäten leidet VENOM unter folgenden grundsätzlichen Problemen (teilweise werden an anderer Stelle beschrieben):

- Signif. Steigerung von Entwicklungs- und Betriebskosten
 - **Hohe Aufwände zur Entwicklung** und Inbetriebnahme neuer Features. Teilweise dauert es auch bei wichtigen Anforderungen mehr als 4-6 Monate, bis die entsprechende Funktion in Betrieb gehen kann.
 - Hohe Fehleranfälligkeit bei Änderungen und Neuentwicklungen: Selbst kleine Änderungen ziehen substantielle Test- und Bugfixing-Aufwände nach sich.
- Sinkende Betriebsstabilität: Die Anzahl an Fehlern im laufenden Betrieb (Prio-1 Produktionsfehler) nimmt zu.
- Die Verlässlichkeit von Aufwandsschätzungen der Entwicklungsteams nimmt ab.
- Mangelhafte Qualität von Dokumentation

Gefällt mir Sei der Erste, dem dies gefällt. Keine Stichwörter

Schreiben Sie einen Kommentar...



Zusammenarbeit

The screenshot shows a web application interface for a Wiki. The top navigation bar includes a menu icon, the 'arc' logo, and tabs for 'Bereiche' and 'Personen'. A blue 'Erstellen' button is visible. The left sidebar contains a 'BEREICHsverknüpfungen' section with a link to connect content, and a 'SEITENhierarchie' section with a tree view of pages including '1. Einführung', '1.1. Das Unternehmen', '1.2. Das VENOM-System', '1.3. Aufgabenstellung', '1.4. Qualitätsziele', '1.5. Stakeholder', '2. Randbedingungen', '3. Kontextabgrenzung', '4. Lösungsstrategie', '5. Bausteinsicht', '6. Laufzeitsicht', '7. Verteilungssicht', '8. Konzepte', '9. Entwurfsentscheidungen', and '10. Qualitätskriterien'. The main content area displays a page titled 'Abbildung 1. Historie von VENOM'. The text describes the challenges of the VENOM system, such as high development costs, low reliability, and poor documentation. A feedback dialog box is open on the right, showing a profile for 'Ralf D. Mueller' and a text input field containing 'Feedback!'. Below the main text, there is a 'Gefällt mir' button, a 'Keine Stichwörter' button, and a comment input field with a placeholder 'Schreiben Sie einen Kommentar...'. The bottom of the sidebar has a 'Bereich konfigurieren' button.





Überarbeiten in PDF

- ... geht fast besser, als in Word



Broken Links

- ...immer wieder mit Probleme bei Dateinamen oder Hyperlinks
- Lösung: automatisierte Tests!



Automatisiertes Testing der Doku

- Broken Cross References (aka Broken Internal Links)
- Missing Images Files
- Multiple Definitions of Bookmarks or ID's
- Missing Local Resources
- Missing Alt-tags in Images



<https://github.com/aim42/htmlSanityCheck>



Automatisiertes Testing der Doku

HTML Sanity Check Results

2	385	10	0.538sec	97% successful
pages	checks	issues	duration	

Results by Page

Page	Checks	Findings	Success rate
index-orga.html	128	9	92%
arc42-venom.html	257	1	99%

Results for index-orga.html

location : C:\Users\ralfd\projects\jax2017\venom-example-bitbucket\build\docs\html5\index-orga.html

78.3	128	9	92% successful
kByte	checks	issues	

Broken Internal Links Check

68 href checked, 2 missing id found.

- link target "IT-Betrieb" missing, reference count: 2 (Suggestions: _relevante_it_prozesse)

Missing alt-attribute declaration in image tags

14 image tags checked, 0 missing alt attributes found.

Duplicate Definition of id Check

37 id checked, 0 duplicate id found.

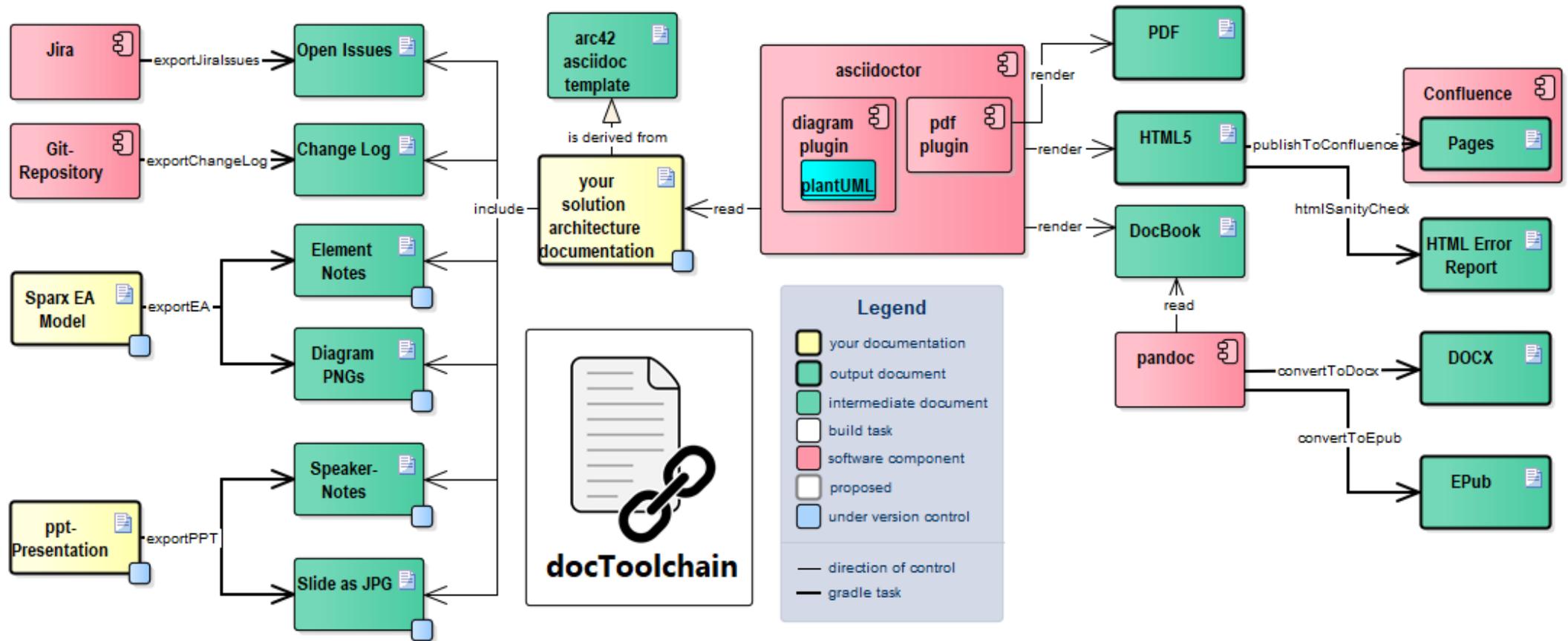
Consistency of ImageMaps



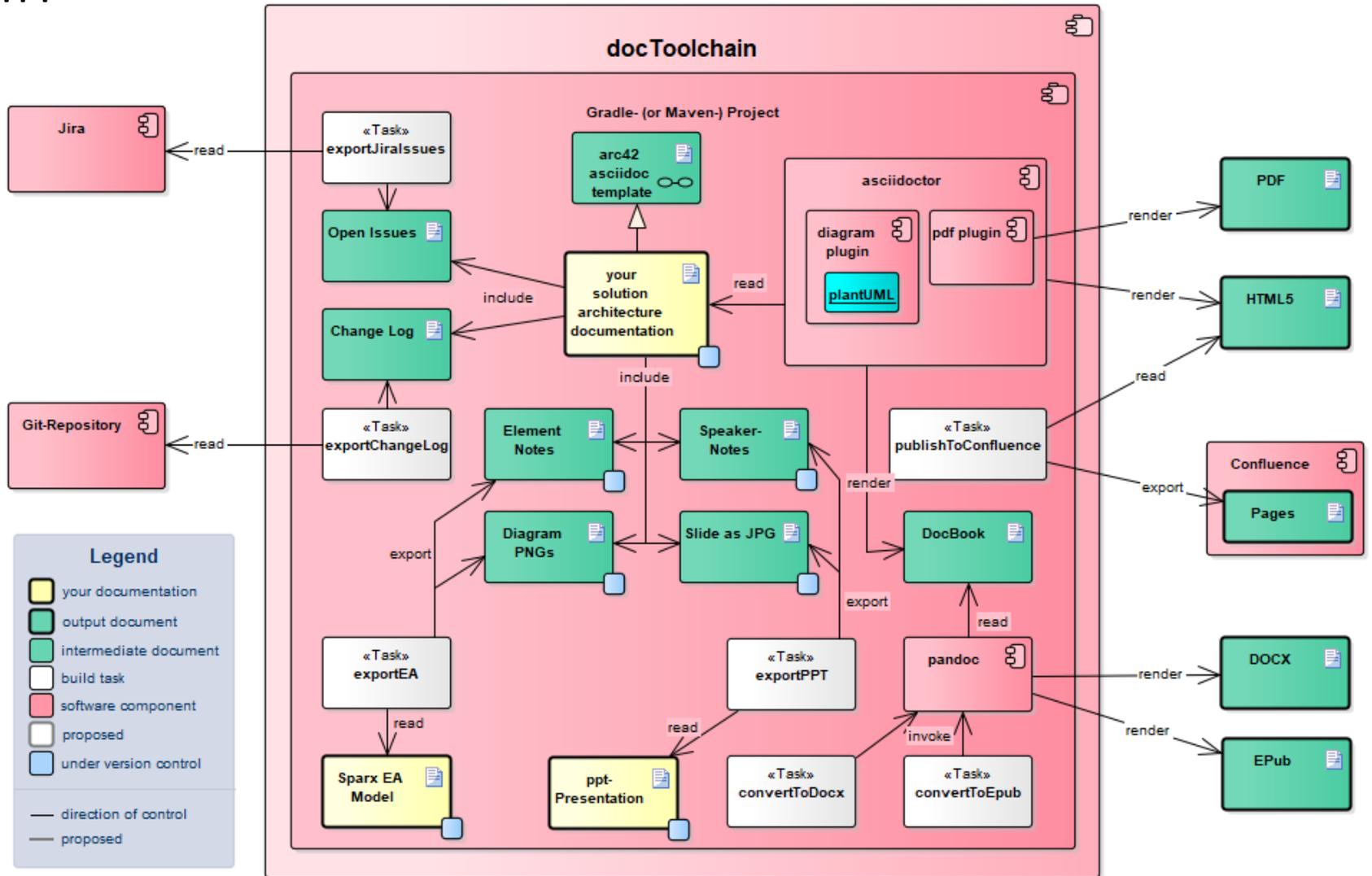
<https://github.com/aim42/htmlSanityCheck>



docToolchain



docToolchain



PDF-Header



VENOM: Architecture Documentation

Author: Gernot Starke @gernotstarke

Über dieses Beispiel

Diese Dokumentation beschreibt die Architektur eines (fiktiven) Softwaresystems. Es dient als Beispiel für:

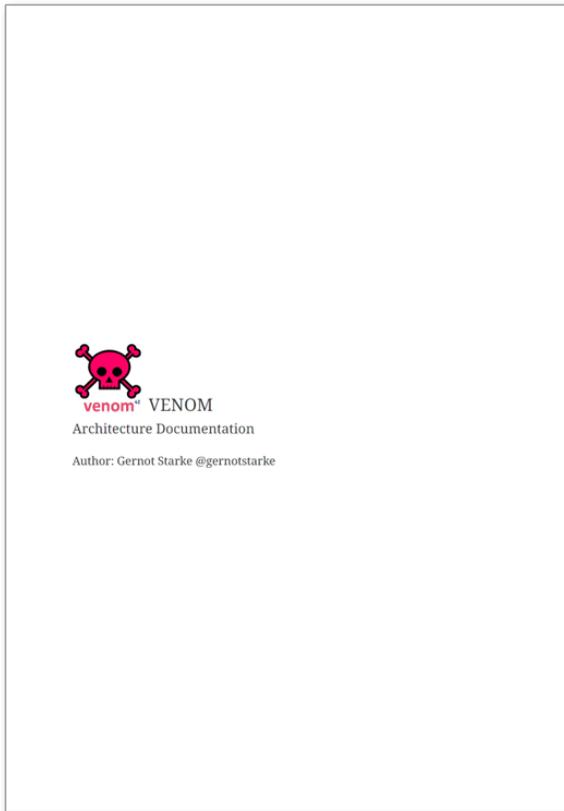
- Architekturdokumentation gemäß [arc42](#)
- Systematische Verbesserung oder Evolution von Systemen gemäß [aim42](#)



Ähnlichkeiten mit bestehenden Systemen sind rein zufällig und nicht beabsichtigt.



PDF-Header



PDF-Header



venomst VENOM
Architecture Documentation
Author: Gernot Starke @gernotstarke



venomst VENOM Architecture Documentation

Über dieses Beispiel

Diese Dokumentation beschreibt die Architektur eines (fiktiven) Softwaresystems. Es dient als Beispiel für:

- Architekturdokumentation gemäß [arc42](#)
- Systematische Verbesserung oder Evolution von Systemen gemäß [aim42](#)



Ähnlichkeiten mit bestehenden Systemen sind rein zufällig und nicht beabsichtigt.





PDF-Header

```
header:                               custom-theme.yml
  height: 0.75in
  line_height: 1
  recto_content:
    right: 'image:logo.jpg[width="30"]'
    center: 'VENOM: {document-subtitle}'
  verso_content:
    left: 'image:logo.jpg[width="30"]'
    center: 'VENOM: {document-subtitle}'
```



PDF-Header



venomst VENOM
Architecture Documentation
Author: Gernot Starke @gernotstarke



venomst VENOM Architecture Documentation

Über dieses Beispiel

Diese Dokumentation beschreibt die Architektur eines (fiktiven) Softwaresystems. Es dient als Beispiel für:

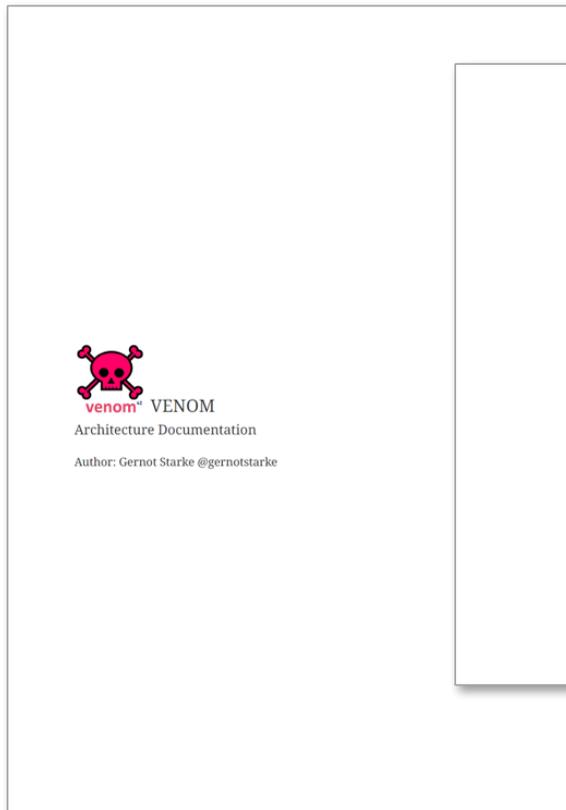
- Architekturdokumentation gemäß [arc42](#)
- Systematische Verbesserung oder Evolution von Systemen gemäß [aim42](#)



Ähnlichkeiten mit bestehenden Systemen sind rein zufällig und nicht beabsichtigt.



PDF-Header



VENOM: Architecture Documentation



Über dieses Beispiel

Diese Dokumentation beschreibt die Architektur eines (fiktiven) Softwaresystems. Es dient als Beispiel für:

- Architekturdokumentation gemäß [arc42](#)
- Systematische Verbesserung oder Evolution von Systemen gemäß [aim42](#)



Ähnlichkeiten mit bestehenden Systemen sind rein zufällig und nicht beabsichtigt.





Seitenumbrüche

- ...machen keinen Sinn für Single-Page HTML
- ...aber für PDF!
- <<<<



Danke!



Gernot Starke

@GernotStarke

<http://arc42.org/>

@arc42Tipps



Ralf D. Müller

@RalfDMüller

<https://rdmueller.github.io>



@docToolchain

Clipart: presentermedia.com, licenced to ralf.d.mueller@gmail.com

