

Wer braucht schon Dokumentation?
Praxisvorlesung TU Dresden
19. Januar 2015

parson
Dienstleister für
Technische Kommunikation
und Wissensmanagement

- Was wir machen
- 1 Technische Dokumentation für Benutzer und Entwickler
 - 2 Prozessoptimierungen
 - 3 Wissensmanagement
 - 4 E-Learning
 - 5 Schulungen, E-Learning und Tutoriel
 - 6 Beratung



Quellen

- [1] [2] [3] [4] [5] [6] [7] [8] [9] [10] [11] [12] [13] [14] [15] [16] [17] [18] [19] [20] [21] [22] [23] [24] [25] [26] [27] [28] [29] [30] [31] [32] [33] [34] [35] [36] [37] [38] [39] [40] [41] [42] [43] [44] [45] [46] [47] [48] [49] [50] [51] [52] [53] [54] [55] [56] [57] [58] [59] [60] [61] [62] [63] [64] [65] [66] [67] [68] [69] [70] [71] [72] [73] [74] [75] [76] [77] [78] [79] [80] [81] [82] [83] [84] [85] [86] [87] [88] [89] [90] [91] [92] [93] [94] [95] [96] [97] [98] [99] [100]





parson

Dienstleister für
Technische Kommunikation
und Wissensmanagement

Was wir machen



Technische Dokumentation für Benutzer und Entwickler



Prozessdokumentation



Wissensmanagement

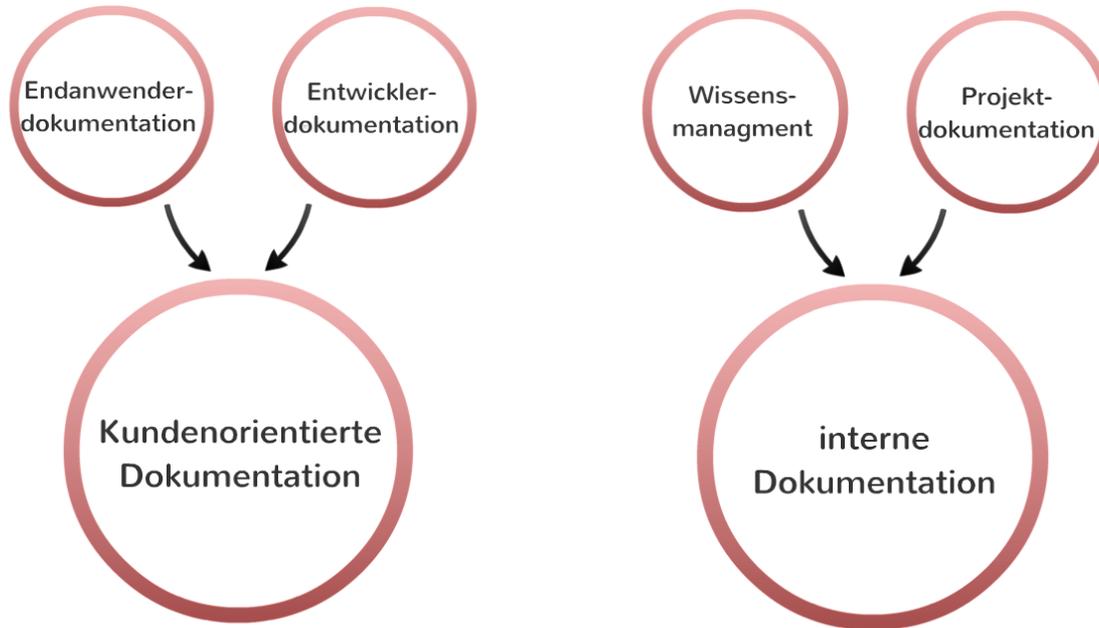


Entwicklung



Schulungen, E-Learning und Tutorials
Beratung

Arten von Software-Dokumentation



Entwickler- dokumentation

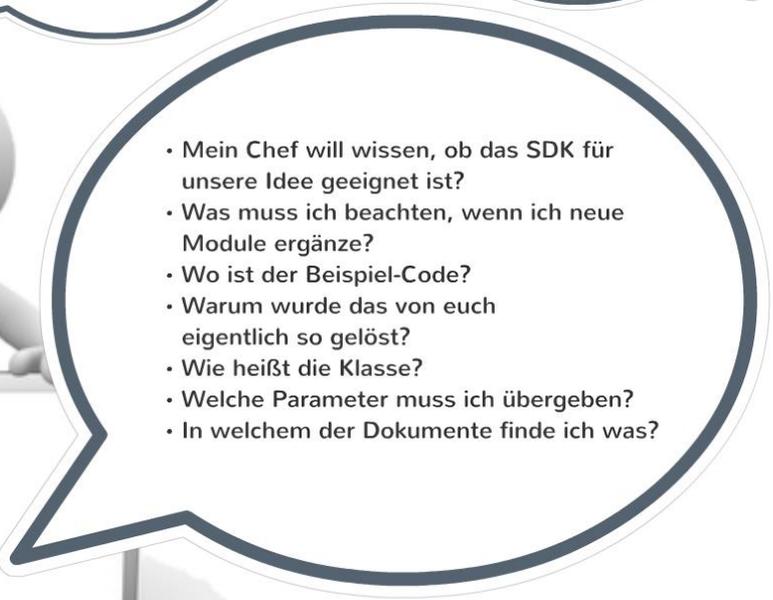
- **Methoden/Klassenreferenz**
- **Architektur-Dokumentation**
- **Fachliche Konzepte**
- **Developer Guide**
- **Tutorials**
- **Beispiel-Code/-Programme**
- **Release Notes**



Das SDK ist an sich gut, aber... die Doku ist *!%%!#*



Ihr nutzt jetzt unser tolles SDK zur Berechnung der KFZ-Steuer?

- 
- Mein Chef will wissen, ob das SDK für unsere Idee geeignet ist?
 - Was muss ich beachten, wenn ich neue Module ergänze?
 - Wo ist der Beispiel-Code?
 - Warum wurde das von euch eigentlich so gelöst?
 - Wie heißt die Klasse?
 - Welche Parameter muss ich übergeben?
 - In welchem der Dokumente finde ich was?



das SDK für
ich neue

Ihr nutzt jetzt unser
tolles SDK zur
Berechnung der
KFZ-Steuer?



Wir schauen L...
Dann überlegen L...
benötigt werden, z...
Referenz. Das geht s...
nach DITA. Am liebsten
Wiederverwendung, Varia...
Source-Publishing für profes...

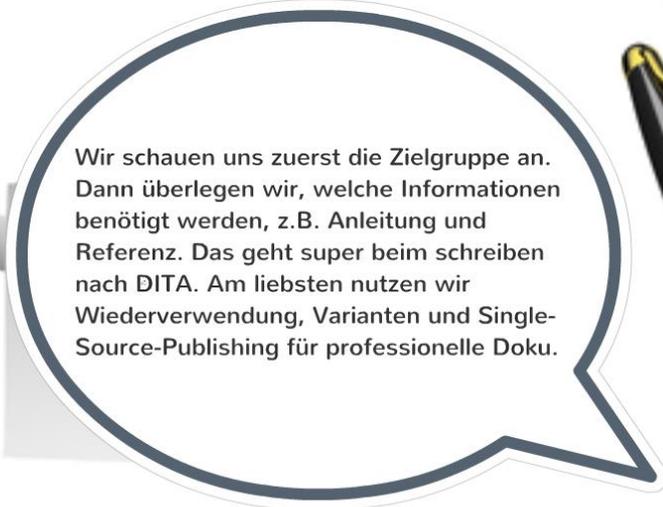
- Neues Vorgehen?
- Zielgruppenanalyse
 - Anwendungsfälle
 - Informationstypen
 - Tools/Formate
 - Publizieren



Wie sieht denn
professionelle Doku aus?
Wie macht ihr als
Redakteure das?



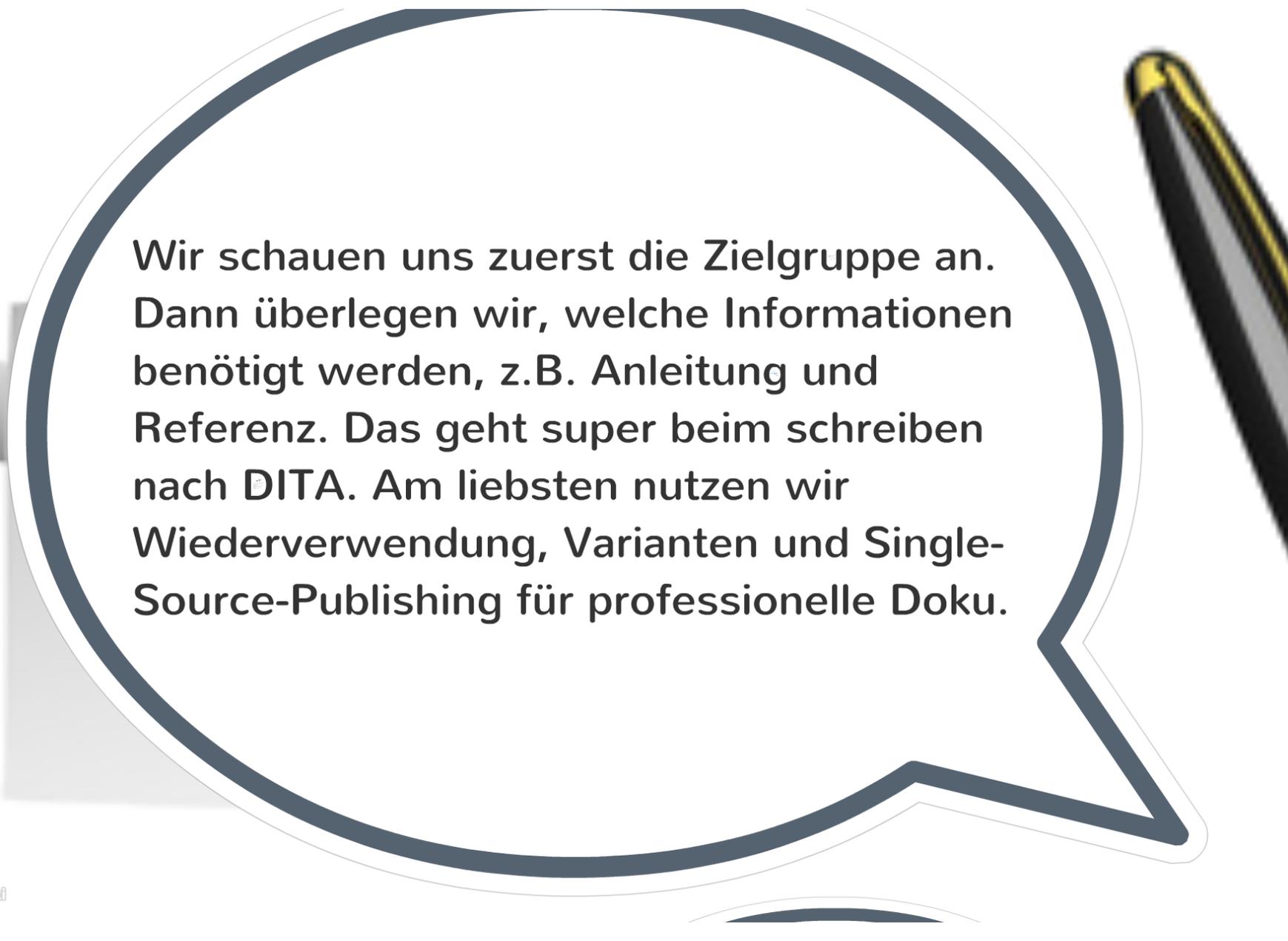
Wir Redakteure sind am
liebsten Teil des
Entwicklerteams, nah an der
Quelle. Praktisch ist es,
wenn die Doku Bestandteil
des automatischen Builds ist.



Wir schauen uns zuerst die Zielgruppe an.
Dann überlegen wir, welche Informationen
benötigt werden, z.B. Anleitung und
Referenz. Das geht super beim schreiben
nach DITA. Am liebsten nutzen wir
Wiederverwendung, Varianten und Single-
Source-Publishing für professionelle Doku.

Neues Vorgehen?

- Zielgruppenanalyse
- Anwendungsfälle
- Informationstypen
- Tools/Formate
- Publizieren



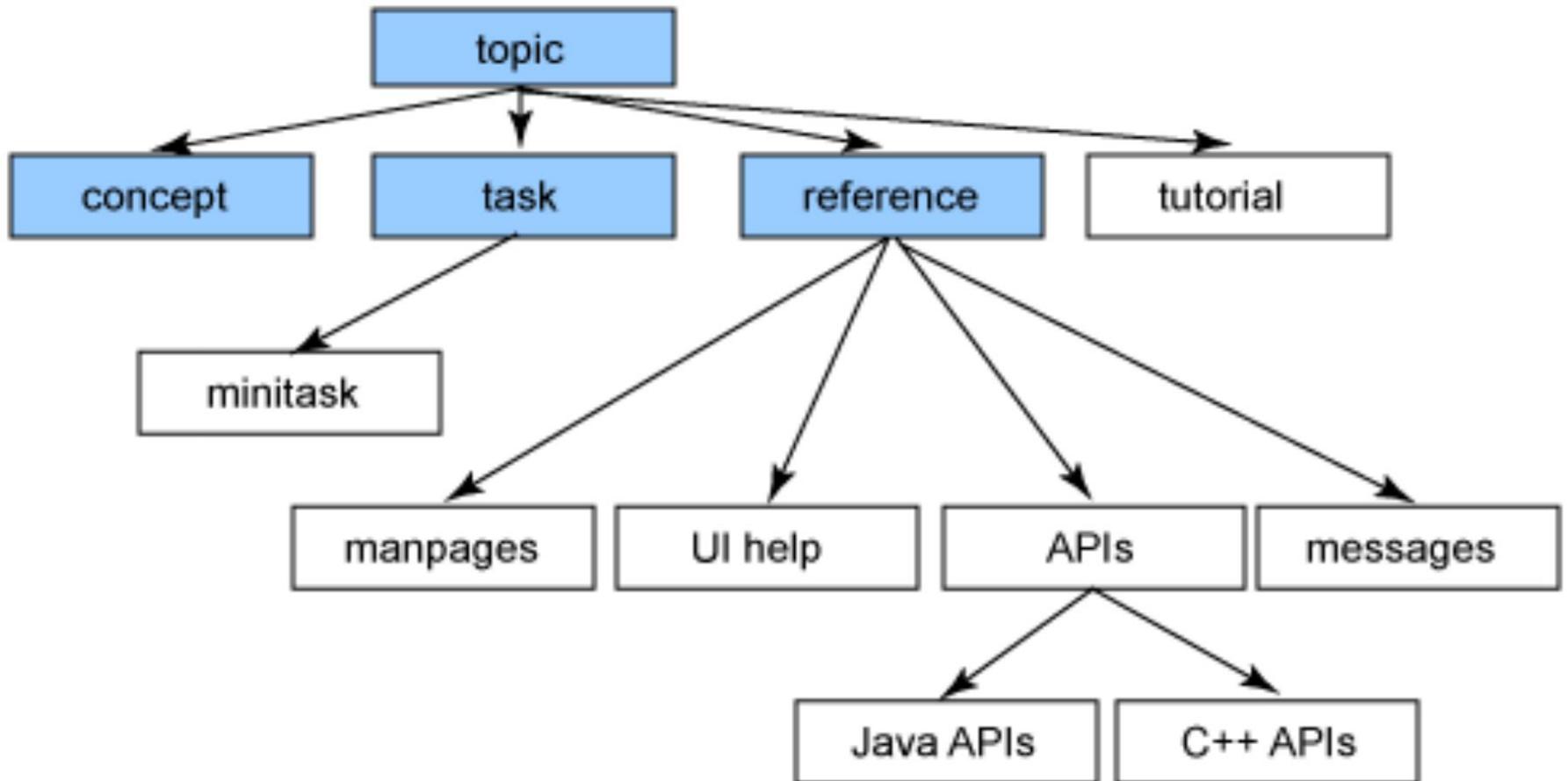
Wir schauen uns zuerst die Zielgruppe an. Dann überlegen wir, welche Informationen benötigt werden, z.B. Anleitung und Referenz. Das geht super beim schreiben nach DITA. Am liebsten nutzen wir Wiederverwendung, Varianten und Single-Source-Publishing für professionelle Doku.

Persona	Behaviour
 Samantha	Starts programming immediately and learns by trial and error. She uses the code comments of the imported classes as documentation. Likes copying code from examples.
 George	Likes reading conceptual information about the architecture and business/technical processes before he starts programming. Consults the documentation to find a solution for a specific <u>task</u> or problem.
 Nic	Reads on demand, that is, whenever he needs specific information regarding a task or a problem.



- Darwin Information Typing Architecture
- XML-Strukturdefinition für Technische Dokumentation
- Topic-basiert
- optimal für modulare Dokumentation, Wiederverwendung, Verlinkung, Varianten
- DITA OT zum Publizieren, z.B. nach HTML und PDF

DITA Topic Typen



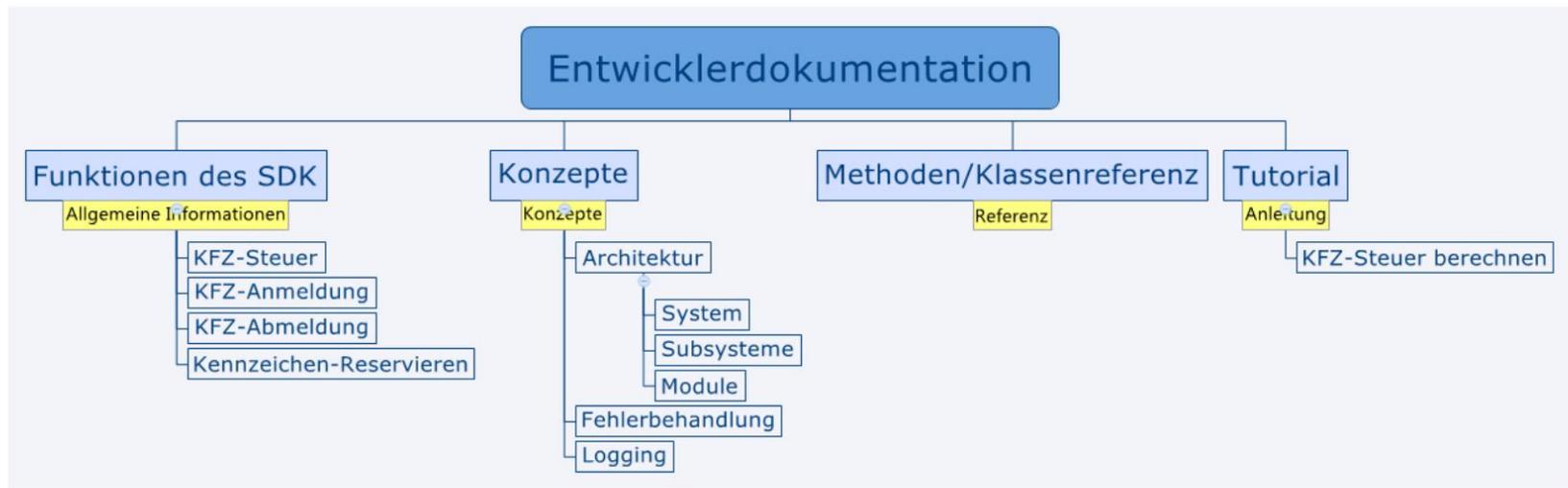
Neues Vorgehen?

- Zielgruppenanalyse
- Anwendungsfälle
- Informationstypen
- Tools/Formate
- Publizieren

Persona: Entwickler Tobi



- Diplom-Informatiker
- C#-Experte
- Web-Anwendungen
- Liest zunächst Konzepte und Architektur, bevor er losprogrammiert.
- Nutzt gerne Beispiel-Code als Ausgangspunkt für eigene Arbeiten.
- Ist von seinem Chef genervt, der kein Entwickler ist und ständig wissen will, was Tobi so macht.
- Hat die Aufgabe die Berechnung der KFZ-Steuer in der APP per SDK zu realisieren.



Doku-Tool für Methoden/Klassenreferenz:

- *Doxygen*
- Javadoc
- [http://en.wikipedia.org/wiki/
Comparison_of_documentation_generators](http://en.wikipedia.org/wiki/Comparison_of_documentation_generators)

Allgemeine Informationen, Konzepte, Anleitungen:

- *XML-Editor*
- Word
- Wiki
- Doku-Tool (ergänzt per Markdown Language)

Markup Language vs. Markdown Language

person Developer Documentation Showcase - SDK KFZ 4 You

Content Search

- Über KFZ 4 You
- Grundlegende Konzepte
- Methoden/Klassenreferenz
- Packages
- Liste aller Klassen
- Index aller Klassen
- Hierarchie der Klassen**
- Alle Bestandteile
- Funktionen
- Variablen
- Dateien
- Tutorials

KFZVerwaltung
Fahrzeug

[Public Member Functions](#) | [Protected Attributes](#) | [List of all members](#)

KFZVerwaltung.Fahrzeug Class Reference

[More...](#)

Inheritance diagram for KFZVerwaltung.Fahrzeug:

```
graph BT; KFZVerwaltung_SaisonFahrzeug[KFZVerwaltung.SaisonFahrzeug] --> KFZVerwaltung_Fahrzeug[KFZVerwaltung.Fahrzeug];
```

Public Member Functions

- [Fahrzeug \(\)](#)
- [Fahrzeug \(int hubraum, string kennzeichen\)](#)
- string [ErmittleKennzeichen \(\)](#)
- virtual int [BerechneKFZSteuer \(\)](#)
- override string [ToString \(\)](#)

Protected Attributes

- int [m_Hubraum](#)
- string [m_Kennzeichen](#)

Detailed Description

Die Klasse [Fahrzeug](#) dient zur Verwaltung eines Fahrzeuges. Sie speichert den Hubraum und das dazugehörige Kennzeichen. Des Weiteren berechnet sie die KFZSteuer für das [Fahrzeug](#).

Constructor & Destructor Documentation



we create knowledge

parson



Quellen

- <http://dita.xml.org>
- <http://www.ibm.com/developerworks/library/x-dita2/>
- <https://www.parson-europe.com/de/wissensartikel/11-documentation-for-software-engineers.html>
- © Anatoly Maslennikov / fotolia.com
- © coramax / fotolia.com

parson

Anne Hoffmann

contact@parson-europe.com

040 7200 500 0