

# Die Magie hinter Dexterity

Dexterity und Behaviors unter die Haube geschaut.



**Wann**

11.03.2020 von 09:30 bis 09:55 (Europe/Berlin / UTC100)



**Termin zum Kalender hinzufügen**

[iCal](#)

**Kurzvortrag (25 min)**

**Raum:** Fehlender Wert: Raum I88

**Zielgruppe:**

KFZ-Meister zum Lehrling: "Und jetzt üben wir das Kopfschütteln beim Öffnen der Motorhaube".

Beim Öffnen des Dexterity-Codes passiert einem das Kopfschütteln vielleicht erstmal einfach so – obwohl der Code inzwischen sehr sauber und leserlich ist. Aber es soll ja alles dynamisch (=rechenintensiv), aber doch schön schnell sein, also hat es überall Caches drin, was die Lesbarkeit nicht verbessert. Ich zeige, welche Module am Dexterity-Types-System beteiligt sind.

Es kommen halt viele Dinge zusammen, wenn man flexible, TTW-konfigurierbare Inhaltstypen programmieren will, und genau das ist Dexterity mit seinem Hauptschema (wahlweise XML oder im Code) und den Behaviors (wahlweise zentral oder context-spezifisch). Ich zeige, was die Unterschiede sind und wie man das für sich nutzen kann. Im Speziellen gehe ich auf Behaviors ein, da es hier ein paar interessante, oft unterschätzte Möglichkeiten gibt.

Es wird ein kurzer Spaziergang durch den Plone-Core-Code mit kleinen Beispielen.



## Jens Klein

### [Klein & Partner KG](#)

Der Geschäftsführer der [Klein & Partner KG](#) ist seit 1992 unternehmerisch in der IT tätig. Die Klein & Partner KG hat er 2004 als sein drittes Unternehmen gegründet. Das Python-basierte [freie CMS-Projekt Plone](#) und seine Community begleitet Jens nun schon seit 2003. Er ist dort Kernentwickler und Mitglied des [Board of Directors](#) und des [Security-Teams](#).

Twitter: [@yenzenz](#)

Github: [jensens](#)

Email: [jk@kleinundpartner.at](mailto:jk@kleinundpartner.at)

Inhaltspezifische Aktionen



[PDF](#)

[Zurück: YAFOWIL, eine flexible Alternative zu z3c.form](#) [Weiter: Digitale Zugänglichkeit - ein Pfeiler der nutzerzentrierten Softwareentwicklung](#)