



Nimmer mehr

Wir sind eine Gesellschaft des „immer mehr“. Doch nicht nur in Bezug auf die Umwelt erscheint mir das der falsche Weg.

Ich bin ja ein großer Freund von Visual Studio Code. Wann immer es etwas an Textdateien, Markdown, JavaScript- oder HTML-Code zu tun gilt, verwende ich genau diesen Editor. Und das schon ziemlich lange. Die Version weiß ich nicht mehr, aber der war nur unter ohrenbetäubendem Protest von seinem Schnuller zu trennen. Kleinkindstatus. Trotzdem wurde er von mir sofort als „extrem nützlich“ verschlagwortet und fortan exzessiv verwendet.

Seitdem hat sich viel getan. Meine Haare sind grauer geworden und ich habe immer so ein Ziehen ... aber das ist eine andere Geschichte. Visual Studio Code auf jeden Fall ist gewachsen, älter und reifer geworden. Und zwar enorm. Wo es am Anfang nur die rudimentären Funktionen für das Öffnen, Speichern et cetera gab, werden nun Terminal und die Verwaltung von Dateien angeboten. Jede Menge Intelligenz sorgt für Syntax-Highlighting, Emmet, IntelliSense und vieles mehr.

Um den Vergleich mit dem Menschen noch einmal heranzuziehen, würde ich sagen, der Editor ist erwachsen, in den besten Jahren.

Alle Funktionen, die man benötigt, sind vorhanden.

Das merke ich daran, dass sich beim monatlichen Update immer seltener der Wow-Effekt einstellt: Was für eine tolle, neue, hilfreiche Funktion.

Und was kommt nach dem Erwachsenwerden? Dann lebt es sich erst einmal eine Zeit lang wunderbar. Aber irgendwann kündigt sich der Verfall an. Glieder tun einem weh, graue Haare. Der Mensch altert.

Um das Licht wieder auf Visual Studio Code zu werfen: Statt Begeisterung ob der neuen Funktion jetzt nur Schulterzucken.

Das soll keine Kritik am Entwicklungsteam des Editors sein. Die Jungs und Mädels leisten nicht nur einen hervorragenden Job, sie setzen ja auch viel von dem in der Software um, was sich die Community wünscht.

Aber irgendwann sind eben alle Funktionen drin, die für ein Einsatzgebiet benötigt werden. Mehr führt eher zu Overkill und Verfall.

Es sei denn, die Anforderungen ändern sich und das Projektteam muss die Software nachsteuern – beispielsweise, wenn der Editor plötzlich Mails versenden oder Zeichnungen anfertigen sollte. Für Visual Studio Code hoffe ich, dass es nie so weit kommt. Denn das würde den hervorragenden Editor schädigen.

Viel Spaß mit der neuen dotnetpro

Tilman Börner
Chefredakteur dotnetpro



David Tielke

sagt Ihnen, worauf Sie achten müssen, damit Software wartbar bleibt (S. 45)



Patrick A. Lorenz

automatisiert das Kümmern um die Softwarequalität mithilfe statischer Code-Analyse (S. 110)



Fabian Deitelhoff

stellt den Micro-OR-Mapper Dapper vor und zeigt dessen Grenzen (S. 98)