



Lahme Software

Software ist zu langsam.
Immer. Immer.
Immer.

Die Website der dotnetpro braucht satte zehn Sekunden, bis sie komplett geladen ist? Zu langsam. Visual Studio ist auf dem einen Rechner erst nach sechs Sekunden bereit, die ersten Klicks entgegenzunehmen. Auf dem anderen nimmt sich die Software sogar satte 20 Sekunden heraus. Zu langsam. Ein neues Projekt anlegen? Fast neun Sekunden, bis das WPF-Formular geladen wurde. Viel zu langsam.

Dabei verkünden manche mit stolzeschwellter Brust die überragende Performance, die ihr System an den Tag legt. Da wird beworben, dass ein Webserver 10000 Anfragen pro Sekunde abarbeiten kann. Oh wie toll. Nur: Mir doch egal. 10000 Anfragen richte ich nicht an einen Webserver. Braucht die Website hingegen zehn Sekunden, bis sie geladen ist, dann betrifft mich das direkt. Übrigens: Selbst die Google-Homepage braucht über eine Sekunde, bis sie geladen ist.

Das liegt natürlich meist nicht nur an der Seite oder dem Server, sondern daran, dass noch von anderen Servern Daten wie etwa Banner geladen werden müssen.

Ganz übel aber sind Verzögerungen oder englisch Latency, wenn es um die direkte Interaktion mit dem Computer geht.

Ab einem gewissen Verzug sind Programme nicht mehr bedienbar. Geben Sie zum Beispiel Text über die Tastatur ein und wird dieser erst mit einer Verzögerung angezeigt, ist das nervtötend. Wenn Sie dabei nicht auf den Bildschirm blicken müssen, mag das noch hinnehmbar sein. Wehe aber, Sie müssen den Text verändern. Tritt bei der Navigation im Text eine Verzögerung auf, hat das den Sprengstoff, den Bediener an den Rand des Wahnsinns zu treiben.

Untersuchungen zeigen genau, welche Verzögerungen noch hinnehmbar sind und bei welchen die Mehrheit der Probanden den Daumen nach unten senkt. So sind laut [1] beim Ziehen von Objekten auf dem Bildschirm schon Verzögerungen von zwei Millisekunden bemerkbar, und Verzögerungen von 80 Millisekunden gelten als absolutes Maximum. Auf der Website wird in Videos auch sehr eindrucksvoll gezeigt, was solche Verzögerungen bedeuten. Diese Website lädt übrigens in nur wenig mehr als einer Sekunde. Respekt.

[1] <https://www.inkandswitch.com/slow-software.html>

Viel Spaß mit der dotnetpro

Tilman Börner
Chefredakteur dotnetpro



Tobias Meier

betreibt Microservices mit dem neuen Azure Service Fabric Mesh (S. 60)



Robert Mühsig

stellt die Datenbank ESENT vor, die jeder hat, aber keiner kennt (S. 92)



Tobias Hopenthaler

baut, testet und veröffentlicht mit Visual Studio App Center mobile Apps (S. 106)