

## Gut Ding braucht Ewigkeit



**S**ie kennen diese Geschichte: Damals und lang ist's her – der erste Kontakte mit Rechnern. Da hören Sie dann Begriffe wie C64 oder Z81. Bei mir war es ein CPM und natürlich war Basic die erste Programmiersprache. Ich will gar nicht erzählen, dass ich mein erstes Programm ohne Return-Taste eingegeben habe, was bedeutete, dass der Rechner das Programm gar nicht gespeichert hatte.

Richtig begeistert haben mich dann aber Rechner, die dBase sprachen. Denn schon damals, im zarten Alter von 19 Jahren, war mir

klar, welchen Nutzen die strukturierte Ablage von Daten mit sich bringt. Ich muss mich nicht darum kümmern, wie die Informationen abgelegt werden. Ich muss mir keine Sorgen machen, wie ich die Details wohl wieder herausbekomme. Ich kippe die Informationen hinein und kann über Suchalgorithmen jederzeit alles wieder finden.

Damals sind mithilfe von dBase III+ richtige Verwaltungsanwendungen entstanden, die etwa die Mitglieder meines Rotkreuz-Kreisverbands erfassten. Dabei war das Aufwändigste die Programmierung der Masken – dann nämlich, wenn die nicht nach 08/15 aussehen sollten.

Das war im Jahr 1985. Es ist also 20 Jahre her. Was hat die Softwareindustrie in diesen 20 Jahren eigentlich gemacht? Geschlafen? Geträumt? In der Nase gebohrt? Ich wiederhole es noch einmal: dBase, keine Relationalität, keine Transaktionen, aber mit einer Programmiersprache, mit der man die meisten Verwaltungsaufgaben ganz gut lösen konnte.

Und was hat sich in der Zwischenzeit getan? SQL hat sich als Schnittstelle durchgesetzt. Aber können Sie sich noch an die ersten Kontakte zwischen Visual Basic und SQL Servern erinnern? Meist endete das Ganze in einem Fiasko, weil der eingebundene ODBC-Treiber über bestimmte Funktionen einfach nicht verfügte.

Microsoft hat in den 20 Jahren außerdem den Datenbankzugriff mindestens 20 Mal geändert. Und jede Version des Zugriffs war natürlich inkompatibel zur vorhergehenden. Ganz zu schweigen davon, dass man neben dem ganzen Gedöns mit den Zugriffstechnologien auch noch fließend SQL sprechen musste – und zwar jeweils den SQL-Dialekt, den die angebundene Datenbank sprach.

Und heute? Wie sieht es heute aus? Jetzt kommt Microsoft daher und verkündet LINQ. Sapperlot, welch tolle Neuerung: Jetzt wird der Zugriff auf Datenbanken Teil der Programmiersprache. So schlau das ist, aber hatten wir das nicht schon mal? Ich erinnere mich dunkel an den Anfang meines Editorials. War da nicht was? Damals zu dBase-Zeiten war das alles schon drin. Aber wahrscheinlich ist auch die Softwareindustrie Moden unterworfen. Retro ist in und so schwimmt auch die Softwareindustrie auf der Nostalgiewelle.

Meines Erachtens ist es völlig unverstündlich, dass solche wichtigen Funktionen nicht vor zehn Jahren schon Teil von Visual Basic 1.0 waren. Denn auch wenn die Notwendigkeit einer strukturierten Speicherung von Daten zugezogen hat – benötigt haben wir Programmierer von damals sie auch schon. Ganz zu schweigen davon, dass sie auch exzessiv verwendet worden wäre.

Aber okay, na gut. Werfen wir unser SQL-Wissen über Bord und lernen die neuen Sprachkonstrukte in C# 3.0. Mit etwas Glück können wir auch in C# 10.0 noch darauf aufbauen. Vielleicht aber kommen inzwischen DAO oder ADO zurück, als DAO.NET oder ADO.NET – hmm, aber gab es eines von den beiden nicht schon? Egal.

Ich wünsche Ihnen viel Spaß mit der dotnetpro. Wir bieten Ihnen in dieser Ausgabe jedenfalls wieder ein breites Themenspektrum von Architektur über Praxis bis hin zu Aktuellem und Grundlagen.

Viel Spaß beim Schmökern wünscht Ihnen

Tilman Börner  
Chefredakteur dotnetpro