

Denkanstöße 11 - Am Anfang war das Bild

Der Erfolg der Softwareentwicklung hängt vor allem davon ab, wie gut wir es schaffen, Gedankenmodelle auf technische Systeme zu übertragen. Grafische oder modellbasierte Programmierwerkzeuge sind hierbei eine wertvolle Hilfe.

Leider ist der Würfel in unserem Kopf oft sehr einseitig abgenutzt und fällt gerne immer wieder auf dieselben Ideen und Gedankenmuster. Im Gegensatz zum regulären Glücksspielwürfel ist das zwar zulässig, aber nicht unbedingt wünschenswert.

Stellen Sie sich vor, Sie müssen das Gedankenmodell einer Lösung auf eine Elektronik abbilden, indem Sie beispielsweise Assembler, C oder C++ programmieren. Wie gehen Sie dabei üblicherweise vor? Sie denken nach, bis eine Idee vor Ihrem geistigen Auge auftaucht. Dann nehmen Sie vielleicht ein Blatt Papier und zeichnen Kästchen, Kreise und Pfeile darauf. Das alles wird dann irgendwann in einer Folge von Codezeilen und Datenstrukturen verschlüsselt. Verschlüsselt insofern, als die reale Welt praktisch nicht mehr wiederzuerkennen ist.

Was die Aufgabenstellung betrifft, steckt die Kreativität in der Beschreibung der Lösung. Alles andere ist ein Übersetzungsvorgang, um auf einer Elektronik, die anders als der Mensch „denkt“, die Lösung zu implementieren. Das ist äußerst uneffektiv, weil unser Gehirn nicht in Codezeilen, sondern in zeitlichen, räumlichen und emotionalen Modellen der realen Welt denkt.

Beim Codieren in C oder gar Assembler muss das Gehirn ständig die große Kluft zwischen diesen Gedankenmodellen und einer Maschinensprache überbrücken. Es hat dieselben Nachteile wie die Kommunikation über den Umweg einer Fremdsprache: Es geht langsamer, es treten häufiger Fehler auf und in vielen Fällen werden faule Kompromisse geschlossen, weil die passenden Worte und Formulierungen nicht geläufig sind. Sicherlich lässt sich hier durch Übung vieles beschleunigen und verbessern, doch es stellt sich die Frage, ob es die Mühe wert ist, wenn wir genauso gut unsere Muttersprache benutzen können.

Bereits ein Baby beherrscht schon sehr gut die Interpretation von visuellen Eindrücken. So kann es schon sehr früh zwischen bekannten und unbekannten Personen unterscheiden und Mimik verstehen. Unsere Sprache funktioniert nur, weil sich Worte und Sätze vor unserem geistigen Auge in Bilder und Geschichten verwandeln. Der Mensch kann nur in Worte fassen, was er sich irgendwie vorstellen kann. Das Wort ohne passendes Bild und der Satz ohne vorstellbare Geschichte sind für unser Gehirn nur bedeutungslose Ton- oder Zeichenfolgen.

In Abwandlung eines biblischen Spruches gilt tatsächlich: „Am Anfang waren das Bild und die Geschichte.“ Lange bevor wir eine Sprache lernen und sehr lange bevor wir Programmiersprachen anwenden, sind wir bereits Spezialisten in der Nutzung und Interpretation visueller Signale. Die Konsequenz für die Wahl einer Programmiersprache ist somit ganz simpel: Je näher sie an der dem Menschen seit frühester Kindheit vertrauten bildlichen Vorstellungswelt liegt, umso effektiver ist sie.

Vergleichen Sie einfach einmal die Geschwindigkeit, mit der Sie Cartoons lesen, mit dem Lesen eines Romans. Im Gegensatz zum Roman können wir beim Cartoon die Sprachblase analytisch lesen und gleichzeitig die Situation im Bild intuitiv erfassen. Es ist praktisch unmöglich, dieselbe Informationsdichte nur über Text so effektiv und so eindeutig zu transportieren. Der Nachteil ist, wie wir alle wissen, dass uns der Freiraum für eigene Fantasien und Bilder genommen wird. Aber das ist im Falle der Softwareentwicklung mit „Cartoon-Programmiersprachen“ wie UML eher ein Vorteil.

„Cartoon-Programmiersprachen“ fördern auch die Kommunikation im Team und erleichtern so die Zusammenarbeit. Schließlich fordern grafische Simulationen die Neugierde und den Spieltrieb heraus. All das wirkt stimulierend auf unser Belohnungssystem im Gehirn und ist damit eine treibende Kraft für motiviertes, kreatives und effektives Denken und Handeln.

Na, wenn das keine guten Gründe für grafische Programmiersprachen sind!

Ich freue mich auf Ihre Denkanstöße unter denkanstoss@microconsult.de.

Peter Siwon

Test: Gedankenmodelle

Denken Sie kurz über folgende Fragen nach:

- Wenn Sie einen Roman lesen, was sehen Sie vor Ihrem geistigen Auge, den Text oder eine Geschichte mit Bildern?
- Wenn Sie ein Seminar besuchen dürften und hätten die Wahl zwischen Ihrer Muttersprache oder einer Ihnen geläufigen Fremdsprache, welche Sprache würden Sie bevorzugen?
- Was ist der Grund für Ihre Entscheidung bei der Wahl der Seminarsprache?
- Wenn Sie eine Idee für eine Softwareapplikation haben, wie teilen Sie diese Ihren Kollegen mit?