



Neulich beim Apéro

Wenn ich an Apéros oder ähnlichen Anlässen gefragt werde, was ich denn so treibe an der Uni, pflege ich wahrheitsgemäss zu antworten, dass ich dort eine Forschungsgruppe für Requirements Engineering leite. Nicht selten kommt dann die vorsichtige Frage zurück, was denn das sei. Nun habe ich zwei Optionen. Ich kann zum Monolog über Anforderungen, deren Gewinnung, Modellierung und Validierung ausholen. Je nach gewähltem Fremdwörteranteil eignet sich das vorzüglich, um das Gespräch mit weniger sympathischen Zeitgenossen relativ bald und schmerzlos zu Ende zu bringen. Interessanten Gesprächspartnern antworte ich dagegen viel lieber mit nur einem Satz:

«Verstehen und beschreiben, was mein Kunde will und braucht.» Diesen Satz finde ich deswegen spannend, weil er mehr Fragen als Antworten enthält: Wie verstehen wir eigentlich? Was müssen wir beschreiben und wie? Was ist wichtiger, verstehen oder beschreiben? Wer ist eigentlich «mein Kunde»? Wieso «will und braucht» und nicht nur «will»? In diesen Fragen steckt genügend Gesprächsstoff, um aufkommende Langeweile ebenso wie tiefschürfende Diskussionen über die Katze von Bill Clinton (oder war's die Brille von Bill Gates?) zuverlässig zu verhindern.

Nur schon die erste Frage, die nach dem Verstehen, hat es in sich. Ob das eigentlich Lehr- und lernbar ist – und was zu lernen ist. Nämlich nicht primär Frage- und Interviewtechniken, nicht JAD und RAD und auch nicht UML, OOA, OMT, SA* oder wie sie alle heissen. Sondern etwas viel Einfacheres (oder doch Schwierigeres): Zuhören, Geduld, sich einfühlen, Gehörtes mit eigenen Worten wiedergeben, das eigene Verständnis im Gespräch mit den Kunden immer wieder prüfen und kritisch hinterfragen, unterschiedliche Vorstellungen

verschiedener Kunden erkennen und zu einem Konsens führen – mehr Menschenkennerin als Ingenieur. Oder doch nicht? Verstehen heisst auch aufschreiben können, heisst Zusammenhänge, Eigenschaften, Abhängigkeiten erkennen und formulieren, heisst Lücken und Widersprüche erkennen und auflösen. Dafür sind klassische Ingenieurstugenden gefragt: die Fähigkeit zum Analysieren, Strukturieren, Modelle bilden... Hier kommen UML & Co. nun doch zu ihrem Recht, denn natürliche Sprache ist ein weniger geeignetes Mittel zum Aufschreiben von Anforderungen, weil sie zuwenig präzise ist. Und Aufschreiben muss sein: Die Beschreibung der Anforderungen bildet die Grundlage für die spätere Entwicklung. Also doch mehr Ingenieur als Menschenkennerin? Oder eben beides: Wer die Dinge aufschreiben will, muss sie verstehen und wer die Dinge verstehen (und kommunizieren!) will, muss sie aufschreiben.

Nur schade, dass solche Gespräche meistens nicht zu Ende geführt werden können, sondern abrupt mit der Feststellung abbrechen, dass das Apérobuffet in der Zwischenzeit von den übrigen Anwesenden schon weitgehend leergeräumt worden sei und sich sputen müsse, wer noch etwas abbekommen will. Aber vielleicht sehen wir uns ja wieder einmal an einem Apéro.

*Dr. rer. nat. Martin Glinz ist seit 1993
Professor für Informatik an der Universität Zürich.*

* Für Jargon-Unkundige:

Joint Application Design,
Rapid Application Development,
Unified Modeling Language,
Object-Oriented Analysis,
Object Modeling Technique,
Strukturierte Analyse.