

# “Software-Entsorgung”: Vorträge und Podiumsdiskussion

Jochen Ludewig

Institut für Informatik der Universität Stuttgart

Nach der landläufigen Vorstellung vom Programmierer ist dieser unablässig damit beschäftigt, Software zu entwickeln, selbstverständlich *neue* Software. Denn Programme verschleiben nicht; einmal fertiggestellt, stehen sie für unbegrenzte Zeit zur Verfügung, ohne weitere Kosten zu verursachen.

Mit der Realität hat diese Idylle wenig zu tun: Nach verschiedenen Erhebungen (vor allem von B.W. Boehm) sind von drei Programmierern zwei mit der Korrektur, Änderung, Anpassung oder Erweiterung *existierender* Programme beschäftigt. Für diese Aktivitäten haben sich die Bezeichnungen “Pflege” oder “Wartung” (“Maintenance”) durchgesetzt, Schönfärbereien, die die auf *Modifikation* zielenden Tätigkeiten denkbar schlecht charakterisieren.

Im Zuge der “Wartung” werden an der Software Änderungen und Erweiterungen vorgenommen, die mit der ursprünglichen Konzeption eigentlich unvereinbar sind. Dadurch werden die Strukturen unübersichtlich, Bezeichner verlieren ihren Sinn, der Code wird unangemessen lang, und die Dokumentation, soweit überhaupt vorhanden, sagt nichts mehr über den aktuellen Stand der Programme aus. Irgendwann erscheint eine Fortsetzung solcher “Pflege” unverhältnismäßig aufwendig, die Software ist zur “Altlast” geworden, wie ein lecker Öltanker, der durch Reparatur wieder in einen akzeptablen Zustand gebracht oder abgewrackt und ersetzt werden muß.

Diese Maßnahmen sind hier nun, um einen weiteren, in der Umweltdiskussion gängigen Euphemismus zu übernehmen, als “Entsorgung” umschrieben. Natürlich können wir leicht Dateien löschen und Bänder freigeben. Aber die obsoleete Software war nicht nur schwer wartbar, sondern auch nützlich, ihre Funktion muß weiterhin zur Verfügung stehen, und zwar in den meisten Fällen ohne längere Unterbrechung. Dieses Dilemma macht heute in allen Unternehmen, Behörden und Organisationen, die sich seit längerer Zeit des Computers bedienen und selbst Software entwickelt oder angepaßt haben, schwer zu schaffen.

Für die Software-Entsorgung gibt es grundsätzlich drei Möglichkeiten:

- **Strukturverbesserung (“Re-Engineering”)**  
Die Programme werden “aufgeräumt” und in sinnvollen Modulen zusammengefaßt. Dabei werden Bezeichner, Kommentare usw. ergänzt oder korrigiert, so daß die Software schließlich wesentlich übersichtlicher und leichter wartbar geworden ist.
- **Nachdokumentation (“Reverse Engineering”)**  
Da oft nur der Code vorliegt, wird aus diesem die Dokumentation erzeugt, die für die Wartung und Neuimplementierung erforderlich ist.
- **Neuimplementierung**  
Nachdem die Anforderungen an die bestehende Software geklärt und durch die inzwischen gewachsenen ergänzt sind, wird das alte System durch eine Neuimplementierung ganz oder partiell ersetzt.

Bei allen Ansätzen können Sprachen, Methoden, Werkzeuge und Normen den Erfolg der Sanierung wesentlich beeinflussen.

Die Beiträge zu dieser Sitzung kommen direkt aus der Praxis. Sie sollen ein Bild der Situation und der heute verfolgten Lösungsansätze vermitteln. Vielleicht werden dadurch auch Forschung und Lehre vermehrt auf dieses große, bislang kaum systematisch bearbeitete Problem aufmerksam.