

Software Engineering – Blatt 1

Rasmus Diederichsen Felix Breuninger
{rdiederichse, fbreunin}@uos.de

22. Oktober 2014

Aufgabe 1.2: Grundlagen: Software-Krise und Software Engineering (40 Punkte)

1. In den ersten Computersystemen diente Software ausschließlich der Berechnung eines einzelnen Algorithmus oder zumindest einer eng begrenzten Problemklasse. Durch die schnelle Leistungssteigerung dieser Systeme und der Ausweitung der Anwendergruppe von Spezialisten zu gewöhnlichen Menschen ohne weiterführende Computer-Kenntnisse stiegen die Anforderungen an Software. Somit wurde Software vielseitiger, umfassender und setzte komplexere Algorithmen ein. Diesem Wachstum im Rahmen von Umfang, Zahl der beteiligten Entwickler und Komponenten hielt der bislang unsystematische Entwicklungsvorgang von Software nicht stand. Diese fehlerbehaftete Vorgehensweise führte häufig dazu, dass Software nicht zum geplanten Preis oder in der geplanten Zeit, teilweise gar nicht, fertig gestellt werden konnte. Ausserdem war Software dementsprechend ineffizient, schlecht wartbar und nicht portierbar. Diese Fehler in der Funktionalität führten zu teuren oder sogar lebensgefährlichen Resultaten.
2. Nachdem auf einer Tagung des NATO Science Committee 1968 diese Problematik konstatiert wurde, wurde rückte verstärkt die Entwicklung von Werkzeugen und Strategien in den Fokus der Forschung, um diese in den Griff zu bekommen. Da die Ingenieursdisziplin viele ähnliche Probleme bereits gelöst hatte, entstand analog dazu das Forschungsgebiet des Software Engineering. Ziel war es, das Vorgehen bei der Softwareentwicklung auf ein wissenschaftliches und systematisches Fundament zu stellen.
3. Die Metrik der *Lines of Code* wirft einige Probleme auf. So bleibt zu definieren, welche Art von Zeilen in die Kalkulation eingehen (z.B. Leerzeilen oder Kommentare). Die ermittelten Resultate können für ein Programm selber Funktionalität stark variieren abhängig von Programmierstil, verwendeter Bibliotheken, (Höhe der) Programmiersprache und autogeneriertem Code.

Aufgabe 1.3: Objektorientierter Entwurf (60 Punkte)

