

SEMAT – Gollum oder Smeagol?

Software Engineering ist und bleibt eine schwierige Disziplin – eine neue Initiative getragen von bedeutenden Exponenten unserer Branche will Abhilfe schaffen. Der Ansatz ist vielversprechend und wird zurzeit heiss diskutiert.



«SEMAT soll alle wichtigen Methodiken des Software Engineerings in einen praktikablen Rahmen fassen.»

Daniel Liebhart ist Dozent für Informatik an der Hochschule für Technik in Zürich und Solution Manager der Trivadis AG.

SEMAT steht für «Software Engineering Method and Theory» und ist eine Initiative, die im November 2009 mit der Veröffentlichung eines Manifests gestartet worden ist. Unterschrieben haben dieses Manifest eine Reihe von international anerkannten Experten auf diesem Gebiet. Es sind Namen darunter wie Barry Boehm (V-Modell- und COCOMO-Erfinder), Erich Gamma (Gang of Four), Ivar Jacobson (Miterfinder von UML), Philippe Kruchten (hat RUP und 4+1-Schichten entwickelt), Bertrand Meyer (unser aller OO-Pabst), Edward Yourdon (der Vater der strukturierten Analyse) und Richard Mark Soley (OMG Chairman). Das allein sollte schon genügen, um diese Initiative zu beachten, auch wenn sie sicherlich nicht der erste und nicht der letzte Versuch darstellt, das schwierige Problem des Software Engineering zu lösen. Interessant ist auch, dass nach knapp drei Jahren die Initiative Einzug in relevante Publikationen unserer Branche wie dem Communications of the ACM oder dem IEEE Computer gefunden hat.

Das SEMAT-Manifest

Das SEMAT-Manifest ist als «Aufruf zum Handeln» (Call for Action) formuliert und geht davon aus, dass der Bau von Informationssystemen durch unreife Praktiken behindert wird. Eine grosse Rolle dabei spielen Faktoren wie das Überhandnehmen von Moderscheinungen als Lösungsansätze, das Fehlen einer soliden und allgemein akzeptierten theoretischen Basis, die Vielzahl der schlecht unterscheidbaren Methoden, das Fehlen empirischer Erkenntnisse und das Auseinander-

derklaffen von Theorie und Praxis. Dagegen soll ein vollständiges Neuaufsetzen der Disziplin «Software Engineering» auf Basis einer soliden Theorie, bewährter Prinzipien und etablierter Praktiken helfen.

Die Basis für Theorie, Prinzipien und Praktiken werden in einem sogenannten Software Development Kernel beschrieben. Dieser Kernel steht heute als Sammlung von Arbeitsinstrumenten («things to work with») und auszuführenden Arbeiten («Things to do») zur Verfügung, die rund um sogenannte Alphas (Abstract-Level Progress Health Attribute) organisiert sind. Die Arbeitsinstrumente sind nichts anderes als eine einfache Prozesslandschaft, die zwischen Kunde, Lösung und Umsetzung unterscheidet und definiert, wie Opportunität, Stakeholder, Anforderungen, Software, Team und Arbeit zusammenhängen. Interessanter sind die einzelnen auszuführenden Arbeiten, die relativ einfach als Prozesse formuliert sind. So gibt es auf Ebene des Kunden beispielsweise Prozesse wie das Vorsehen verschiedener Lösungsmöglichkeiten oder die Berücksichtigung aller Bedürfnisse der Beteiligten, während auf der Ebene der Lösung die üblichen Prozesse wie Anforderungsanalyse, Umsetzen, Testen oder Betreiben definiert sind. Die Umsetzung hingegen sieht Prozesse wie Arbeitsvorbereitung, Koordination oder Teamunterstützung vor.

Die Roadmap

Mit der Bereitstellung einer neuen abstrakten Basis für das Software Engineering ist es gemäss den Initianten von SEMAT jedoch

noch nicht getan. Auf Basis der Kernels sollen Praktiken, Entwurfsmuster und schliesslich Methoden entwickelt respektive zugeordnet werden, die als Grundlage für den Bau von Informationssystemen gelten sollen. Ein erster Schritt ist die im Februar 2012 an die OMG unter dem Namen «Essence» eingereichte Spezifikation, die sowohl den Kernel als auch eine Sprache für die Beschreibung von Methoden umfasst. Weitere Aktivitäten werden durch Arbeitsgruppen vorangetrieben, die sich um Themen wie Definitionen, Theorie oder auch Assessments kümmern und entsprechende Artefakte erarbeiten. Heute versteht sich SEMAT als Grundstruktur in der Form eines umsetzbaren Kerns. Morgen soll es alle wichtigen Methodiken des Software Engineerings in einen praktikablen Rahmen fassen.

Die Kritik

Ein Frage muss erlaubt sein: Warum soll ausgerechnet diese Initiative mit unserem «No Silver Bullet»-Problem aufräumen? Die Antwort steht noch aus. Stärke der Initiative ist der Versuch, Theorie und Praxis parallel zu entwickeln und bereits einige einfache Arbeitsmittel zur Verfügung zu haben. Das allein verdient unsere Aufmerksamkeit. Ihre Schwäche zeigt sich in der Tatsache, dass SEMAT selbst einen sehr hohen Anspruch hat – nämlich den, eine Basis für alle möglichen Methoden vorzugeben. So etwas ist in der Tat sehr anspruchsvoll, und selbst so grosse und wichtige Organisationen wie das IEEE mit seinem Software Engineering Body of Knowledge (SWEBOOK) haben niemals einen solchen Anspruch formuliert. Da ist also Vorsicht geboten. <