

Unterstützung von Entwurfsmustern im Quelltext durch bedeutungsorientierte Dokumentation

Exposé der Doktorarbeit

Die Bedeutung erweiterbarer und wartbarer Software wächst mit zunehmender Komplexität des Technologiemarktes. Es gibt zahlreiche Ansätze, Entwickler bei der Erstellung derartiger Software zu unterstützen. Ein Ansatz sind Entwurfsmuster. Sie helfen bei der Reduzierung des Problems des Architekturzerfalls, der durch naturgemäß steigende Entropie und Komplexität während des Software-Entwicklungsprozesses entsteht. Die Anzahl der dokumentierten Entwurfsmuster nimmt durch neue Publikationen stetig zu. Die Auswahl geeigneter Muster für einen Anwendungsfall wird dadurch immer schwieriger. Weiterhin wird die in der Praxis nur bedingt verfügbare und unzureichende Unterstützung für Entwickler bei der Arbeit mit Entwurfsmustern der intensiven Beschäftigung in der Forschung mit diesem Thema nicht gerecht.

Diese Arbeit leistet einen Beitrag zur besseren Unterstützung des Entwicklers bei der Arbeit mit Entwurfsmustern. Die formale Dokumentation von Entwurfsmustern wird thematisiert, um anhand dessen die Anwendbarkeit eines Musters für einen objektorientierten Quelltext festzustellen und das Muster mit Hilfe von Quelltexttransformationen werkzeuggestützt anzuwenden. Sie stellt eine Methodik bereit für das Anreichern von Programmtext mit semantischen Informationen, die bei der Musterselektion und -anwendung dienlich sind. Die vorgestellte Methodik ist für den Einsatz in Quelltextrevisionen geeignet, deckt also die Forward- und Reengineering-Phase bei der Software-Entwicklung ab.

Anhand von Vorbetrachtungen und im Speziellen einer Fragebogenauswertung zum Thema Entwurfsmuster wird die folgende Forschungsfrage gestellt:

Wie kann der Entwickler bei der Selektion und Anwendung von Mustern auf Quelltext durch Anreicherung des Quelltextes mit Zusatzinformationen unterstützt werden?

Die Betrachtung bestehender Ansätze zur Musterdefinition bzw. Musterdokumentation, zur Musterselektion, Musteranwendung und Mustererkennung zeigt, dass Entwurfsmuster heutzutage nicht ausreichend durch automatisierte Ansätze abgedeckt werden können. Aus diesen Betrachtungen werden weiter verwendbare Konzepte und Ideen bestehender Arbeiten aufgegriffen sowie vermeidbare Mechanismen benannt. Aus der erweiterten Problemstellung, die insbesondere die Allgemeingültigkeit eines neuen Ansatzes, die spezifische Ausrichtung auf Quelltext und die unterstützten musterbezogenen Aktivitäten Selektion und Anwendung fokussiert, wird ein neues Verfahren entwickelt.

Der neue Ansatz besitzt als Fundament eine formale Musterdokumentation, die als maschinengerechte Spezifikation von informalen Musterbeschreibungen verstanden werden kann, wie sie hauptsächlich durch Gamma und seine Mitstreiter populär wurden. Zur Gewinnung einer formalen Musterdokumentation wird ein erfahrener und mit einem formal zu dokumentierenden Muster bestens vertrauter Entwickler herangezogen. Er wählt anfangs einen zum Muster passenden Ausgangsquelltext aus, für den die Anwendung des Musters angebracht erscheint. Derartige Quelltexte finden sich etwa in Musterbüchern oder stellen sich bei der täglichen Entwicklertätigkeit heraus. Nun wird das zu dokumentierende Muster manuell angewandt. Dies geschieht de facto durch Transformation des Ausgangsquelltextes in einen sogenannten Zielquelltext, der den

Ausgangs Quelltext samt angewandtem Muster darstellt. In Zuge dessen muss jede transformierte (also veränderte) Stelle mit einer Quelltextannotation versehen werden. Eine Annotation ist ein bestimmter Kommentar, der oberhalb einer beliebigen Anweisung oder Deklaration angebracht werden kann. Eine Annotation dient neben der Kennzeichnung der Transformation weiterhin als Träger einer Motivation für die Transformation, die somit explizit im Quelltext und zwar für die transformierte Stelle, manuell angebracht wird. Annotationen als ein Schlüsselement der Arbeit können prinzipiell als verhaltensneutrale Programmbestandteile verstanden werden, die Intentionen jeder Art transportieren können. Ferner wird das Konstrukt der dualen Annotation entwickelt, das es erlaubt, die Semantik einer Annotation sowohl für den Menschen als auch für die Maschine leicht lesbar zu gestalten.

Aus den durchgeführten Quelltexttransformationen kann nun die Musterdokumentation aufgestellt werden. Zu diesem Zeitpunkt existieren durch die Annotationen bereits formalisierte Motivationen für die Anwendung des Musters, die bisher in informaler Weise bei Gamma vorlagen. Weiterhin wird jede Transformation dadurch formalisiert, dass sie in feinkörnige Refactoring-Operationen aufgeschlüsselt wird. Zusätzlich werden Vor- und Nachbedingungen für jede Transformation definiert, da es prinzipiell nicht möglich erscheint, die Korrektheit einer Quelltexttransformation zu garantieren.

Da bei der Musterselektion und -anwendung verschiedenste Ausgangs Quelltexte vorliegen können, wird die Musterdokumentation durch Verfeinerungsphasen flankiert. Eine solche Phase fordert die Modifikation des Ausgangs Quelltextes und macht durch das Berücksichtigen von Variationen im Quelltext die Musterdokumentation leistungsfähiger. Dazu trägt auch bei, dass sogenannte Isomorphe aufgestellt werden, das sind zu Anweisungen oder Anweisungssequenzen als gleichartig angesehene Entitäten. Annotationen an sich tragen ebenfalls zur Flexibilisierung der Musterdokumentation bei, denn sie abstrahieren von konkreten Programmanweisungen. Ein weiterer Verfeinerungsschritt ist die Modifikation des anzuwendenden Musters selbst, um Mustervarianten abbilden zu können. In der Arbeit wird der Prozess der formalen Musterdokumentation anhand des komplexeren Entwurfsmusters *Observer* Schritt für Schritt demonstriert. Weiterhin wird auf die Muster *Singleton* und *Composite* im Anhang verwiesen, für die jeweils relevante Informationen zur Musterdokumentation geliefert werden.

Die aufgestellte formale Musterdokumentation dient nun dazu, die Musterselektion und -anwendung für einen vorliegenden, annotierten Quelltext durchführen zu können. In der Arbeit wird beschrieben, wie die Annotation eines Quelltextes durchzuführen ist und welche maschinelle Unterstützung hierbei möglich ist. Quelltext wird als Eingabe für die Musterselektion in gleicher Weise annotiert wird im Rahmen der Musterdokumentation. Weiterhin wird durch Fortführung des Beispiels beschrieben, wie die Selektion und Anwendung eines Musters konkret aussieht.

Ferner wird eine Möglichkeit aufgezeigt, wie der vorgestellte Ansatz in den Entwicklungsprozess eingebunden werden kann.

Klaus Meffert

www.klaus-meffert.de