

Semi-automatische Ableitung externer Anpassungsmechanismen für selbst-adaptive Systeme

Benjamin Nagel

s-lab - Software Quality Lab
Universität Paderborn
Warburger Straße 100
33098 Paderborn
bnagel@s-lab.upb.de

Abstract: Vor dem Hintergrund steigender Komplexität gewinnen selbst-adaptive Systeme immer stärker an Bedeutung. Die Realisierung von Selbst-Adaptivität erfolgt durch Frameworks, die Anpassungslogik in externe Mechanismen auslagern können. Durch Erweiterungen können diese Mechanismen an systemspezifische Anforderungen angepasst werden. Die Ableitung dieser Framework-Erweiterungen aus den Anforderungen wird durch bestehende Ansätze nur unzureichend unterstützt. Selbst-Adaptivität wird in Anforderungsbeschreibungen nur implizit spezifiziert und fehlende Automatismen für die Ableitung führen zu einer erhöhten Fehleranfälligkeit. In diesem Beitrag wird ein Ansatz vorgestellt, durch den externe Anpassungsmechanismen semi-automatisch aus Anforderungen abgeleitet werden können.

1 Einleitung und Motivation

Die steigende Komplexität heutiger Softwaresysteme führt zu immer höheren Anforderungen an die Anpassbarkeit und Wartbarkeit dieser Systeme. Um diesen Anforderungen gerecht zu werden, steht die Softwaretechnik vor der Herausforderung Systeme zu entwickeln, die sich selbstständig an verändernde Bedingungen anpassen können. Diese Systeme werden als „selbst-adaptive Systeme“ bezeichnet.

Die Realisierung selbst-adaptiver Systeme kann durch interne oder externe Ansätze erfolgen. Bei internen Ansätzen wird die Anpassungslogik vollständig in die Software integriert und ist somit direkt mit der Anwendungslogik verknüpft. Die enge Verknüpfung erschwert die Wartung und Wiederverwendung der Anpassungslogik. Aktuelle Ansätze wie das Autonomic Computing [KC03] lagern die Anpassungslogik in externe Mechanismen aus. Durch die Expliztheit der Selbst-Adaptivität werden die Erweiterbarkeit und Wartbarkeit verbessert. Frameworks wie RAINBOW [GCS03] und StarMX [AST09] bieten die technische Grundlage für die Realisierung externer Anpassungsmechanismen.

Während die technische Realisierung selbst-adaptiver Systeme durch bestehende Ansätze unterstützt wird, fehlt eine Methodik zur Ableitung externer Anpassungsmechanismen aus den Anforderungen an ein selbst-adaptives System. Die Anpassungslogik beinhaltet nicht nur komplexe Verhaltensbeschreibungen sondern auch Abhängigkeiten zu System- und Kontextinformationen die innerhalb des Entwicklungsprozesses spezifiziert und realisiert werden müssen.

Das in diesem Beitrag vorgestellte Promotionsvorhaben hat das Ziel, einen Ansatz für die semi-automatische Ableitung von externen Anpassungsmechanismen aus den Anforderungen selbst-adaptiver Systeme zu entwickeln. Hierzu werden im folgenden Kapitel die Problemstellung und die relevanten Forschungsfragen des Promotionsvorhabens beschrieben. Kapitel 3 zeigt bestehende Ansätze in Bezug auf die definierte Problemstellung auf. Der Lösungsansatz für die gegebene Problemstellung wird in Kapitel 4 mit dem erwarteten Beitrag dargestellt. Abschließend wird in Kapitel 5 eine Zusammenfassung gegeben.

2 Problembeschreibung

Das Ziel der semi-automatischen Ableitung ist es, mit möglichst geringem manuellem Aufwand einen externen Anpassungsmechanismus zu generieren, der die Anforderungen an die Selbst-Adaptivität des Systems erfüllt. Bestehende Frameworks bieten eine technische Infrastruktur für externe Mechanismen, die durch Erweiterungen an die systemspezifischen Anforderungen angepasst werden können.

Der prinzipielle Ablauf einer Ableitung ist in Abbildung 1 anhand eines konkreten Beispiels dargestellt. In der Analyse werden Ziele, Anforderungen und mögliche Risikofaktoren identifiziert und spezifiziert. Die Anforderungen können durch kontrolliert natürlich-sprachliche Ansätze wie RELAX¹ [WSB09] mit einer Flexibilität gegenüber ihrer Erfüllung versehen werden. Die Selbst-Adaptivität wird in dieser Phase des Entwicklungsprozesses lediglich implizit durch mögliche Alternativen beschrieben.

In dem dargestellten Beispiel werden die Alternativen „textuelle Darstellung“ und „multimediale Darstellung“ und allgemeine Zielstellungen „möglichst performant“ und „möglichst oft multimediale Darstellung“ angegeben. Als Risikofaktor wird eine „niedrige Bandbreite“ identifiziert, die eine Anpassung der Darstellung erforderlich machen kann. Eine detaillierte Spezifikation der erforderlichen Logik über die Auswahl der verschiedenen Alternativen, wird durch bestehende Beschreibungssprachen nicht unterstützt.

¹ Die Konzepte von RELAX wurden in diesem Beispiel auf die deutsche Sprache übertragen.

Im Design werden Framework-Erweiterungen entwickelt, die die Anpassungslogik in formalen, plattformspezifischen Sprachen beschreiben und so von dem jeweiligen Framework automatisch verarbeitet werden können. Exemplarisch wird in Abbildung 1 eine Anpassungstaktik aus [Ch08] dargestellt, die das Umschalten zur textuellen Inhaltsdarstellung beschreibt. Die Spezifikation der Anpassungslogik erfolgt hier in einer plattformspezifischen Sprache und auf einem sehr detaillierten Modellierungsgrad.

Aus der Spezifikation von Selbst-Adaptivität in den Anforderungen und dem Design ergeben sich zwei wesentliche Problemstellungen für die Ableitung der Anpassungslogik. Die fehlende Mächtigkeit bestehender Beschreibungssprachen erschwert die Spezifikation der geforderten Selbst-Adaptivität auf einer plattformunabhängigen Ebene (Modellierungs-GAP). Außerdem fehlt ein methodisches Vorgehen, um aus den Ergebnissen der Analyse und Anforderungserhebung, die Anpassungslogik abzuleiten (Methodisches GAP).

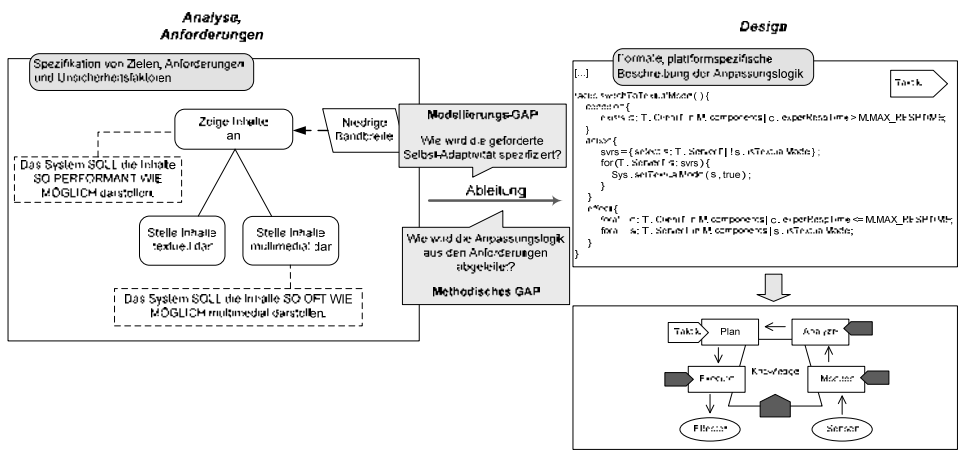


Abbildung 1: Ableitung externer Anpassungsmechanismen

Basierend auf der gegebenen Problembeschreibung, wird für das Promotionsvorhaben die folgende Forschungshypothese aufgestellt: „*Framework-Erweiterungen für externe Anpassungsmechanismen können semi-automatisch aus den Anforderungen selbst-adaptiver Systeme abgeleitet werden.*“. In dem folgenden Kapitel werden bestehende Arbeiten zu der dargestellten Problemstellung aufgeführt und ihre Relevanz für das Promotionsvorhaben aufgezeigt.

3 Verwandte Arbeiten

In der Literatur werden verschiedene Ansätze für die Spezifikation von Anforderungen selbst-adaptiver Systeme aufgezeigt. In [CSB09] werden KAOS Zielbäume [DLF93] mit natürlich-sprachlichen Konzepten von RELAX [WSB09] kombiniert, um Unsicherheiten in der Erreichung von Zielen zu modellieren. Unter Verwendung der Tropos Methodik [BPG04] berücksichtigt [MPP08] auch Abhängigkeiten der Anforderungen zu Elementen in dem System und dem Kontext. In beiden Ansätzen gibt es keine Konzepte, um Selbst-Adaptivität auf einer detaillierten Ebene zu spezifizieren.

In [La03] werden allgemeine, pattern-basierte Ansätze zur Ableitung von Softwarespezifikationen aus Systemzielen beschrieben. Auf Basis von Tropos wurde in [YLL08] eine erweiterte Bibliothek von Pattern für die Ableitung von selbst-adaptivem Verhalten eingeführt. Hierbei werden komponentenbasierte Architekturen abgeleitet, die für eine framework-basierte Realisierung von selbst-adaptiven Systemen zu grobgranular sind. Ein weiterer Nachteil dieser Ansätze ist die fehlende Separierung der Anforderungen an die Selbst-Adaptivität. [ZC06] beschreibt die modellbasierte Ableitung von Petri-Netzen. Dieser Ansatz beschränkt sich auf die Ableitung von Verhaltensaspekten selbst-adaptiver Systeme.

Ein durchgehender Ableitungsansatz von der Anforderungsspezifikation zu konkreten Framework-Erweiterungen, vergleichbar mit dem beschriebenen Vorhaben, ist nicht bekannt.

4 Lösungskonzept

Die Problemstellung und bestehende Ansätze wurden in den Kapiteln 2 und 3 beschrieben. Aufbauend auf den dargestellten Forschungsfragen soll in diesem Kapitel ein Lösungskonzept für die aufgestellte Forschungshypothese skizziert werden. Hierzu wird zunächst der Lösungsvorschlag mit dem erwarteten Beitrag vorgestellt und abschließend die geplante Evaluation des vorgestellten Lösungsansatzes beschrieben.

Die in Kapitel 2 erarbeiteten Forschungsfragen werden durch bestehende Arbeiten in zwei wesentlichen Punkten unzureichend unterstützt:

- 1.) Die explizite und detaillierte Spezifikation von Selbst-Adaptivität im Entwicklungsprozess selbst-adaptiver Systeme.
- 2.) Ein automatisiertes Vorgehen zur Ableitung von Framework-Erweiterungen aus der Anforderungsspezifikation.

Für die Adressierung dieser beiden Punkte, wird in dem Promotionsvorhaben der in Abbildung 2 dargestellte, modellbasierte Ableitungsprozess in Anlehnung an [OMG01] vorgeschlagen.

Für die plattformunabhängige Spezifikation von Selbst-Adaptivität wurde die Modellierungsmethodik *Adapt Cases* entwickelt. Dieser use-case-basierte Ansatz ermöglicht die präzise Spezifikation von Selbstadaptivität auf verschiedenen Abstraktionsebenen. Analog zu „traditionellen“ Use-Cases ermöglichen die *Adapt Cases* eine schrittweise Ableitung der Anpassungslogik aus den spezifizierten Zielen und Anforderungen. Neben der semi-formalen Spezifikation von *Adapt Cases* in der Use-Case Notation, wurde ein Metamodell zur formalen Spezifikation entwickelt, durch das der Kontext, ein Monitoring-Konzept und die Anpassungsaktionen spezifiziert werden können. Als Grundlage für dieses Modell wird das MAPE-K [KC03] Konzept referenziert. Die Semantik der formalisierten *Adapt Cases* wird durch OCL beschrieben. Durch diesen Ansatz wird eine plattformunabhängige Modellierung der Selbst-Adaptivität ermöglicht.

Die formalisierten Modelle der *Adapt Cases* werden in ein plattformspezifisches Modell für ein bestimmtes Framework, z.B. RAINBOW [GCS03] transformiert. Aus diesem Modell können die einzelnen Erweiterungen für das entsprechende Framework durch eine Modell-Text Transformation generiert werden.

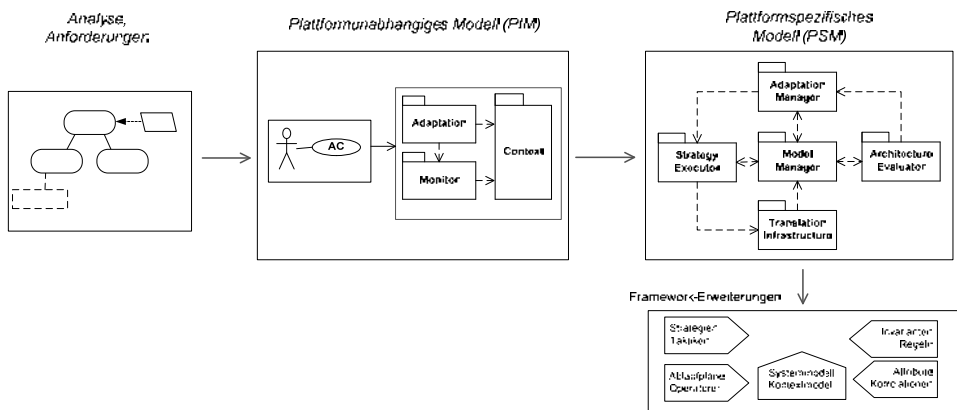


Abbildung 2: Modellbasierter Prozess zur semi-automatischen Ableitung

Der vorgestellte Lösungsansatz soll die in Kapitel 2 beschriebenen Forschungsfragen adressieren. Die einzelnen Ableitungsschritte können durch entsprechende Transformationsregeln automatisiert durchgeführt und so der manuelle Anteil minimiert werden. Hierdurch soll eine wesentliche Reduktion von Fehlern im Ableitungsprozess erreicht werden. Um die Anwendbarkeit des Ansatzes in Softwareprojekten zu ermöglichen, sollen die vorgestellten Konzepte durch ein geeignetes Modellierungswerkzeug unterstützt werden.

Der vorgestellte Lösungsansatz wird im Rahmen des Promotionsvorhabens anhand einer umfangreichen Fallstudie im industrienahen Umfeld des Software Quality Labs (s-lab) evaluiert. Hierdurch sollen die wissenschaftlichen Konzepte validiert und die Anwendbarkeit der Methodik in Entwicklungsprojekten aufgezeigt werden.

5 Zusammenfassung

In diesem Beitrag wurde der Einsatz externer Anpassungsmechanismen für selbst-adaptive Systeme motiviert und Probleme bei der Ableitung dieser Mechanismen aufgezeigt. Es wurde ein konzeptioneller Lösungsansatz aufgezeigt und der erwartete Beitrag in Bezug auf die Problemstellung diskutiert. Als Ausblick wurde die geplante Evaluation im industrienahen Kontext vorgestellt.

Literaturverzeichnis

- [AST09] Reza Asadollahi, Mazeiar Salehie, and Ladan Tahvildari. StarMX: A framework for developing self-managing Java-based systems. *Software Engineering for Adaptive and Self-Managing Systems, International Workshop on*, 0:58–67, 2009.
- [BPG04] Paolo Bresciani, Anna Perini, Paolo Giorgini, Fausto Giunchiglia, and John Mylopoulos. Tropos: An Agent-Oriented software development methodology. *Autonomous Agents and Multi-Agent Systems*, 8(3):203–236, May 2004.
- [CSB09] Betty Cheng, Pete Sawyer, Nelly Bencomo, and Jon Whittle. A goal-based modeling approach to develop requirements of an adaptive system with environmental uncertainty. In *Model Driven Engineering Languages and Systems*, volume 5795 of *Lecture Notes in Computer Science*, pages 468–483. Springer Berlin / Heidelberg, 2009.
- [Ch08] Shang-Wen Cheng. *Rainbow: cost-effective software architecture-based self-adaptation*. PhD thesis, Carnegie Mellon University, 2008.
- [DLF93] Anne Dardenne, Axel van Lamsweerde, and Stephen Fickas. Goal-directed requirements acquisition. *Science of Computer Programming*, 20(1-2):3–50, April 1993.
- [GCS03] David Garlan, Shang-Wen Cheng, and Bradley Schmerl. Increasing system dependability through architecture-based self-repair. In *Architecting dependable systems*, pages 61–89. Springer-Verlag, 2003.
- [KC03] J.O. Kephart and D.M. Chess. The vision of autonomic computing. *Computer*, 36(1):41–50, 2003.
- [MPP08] Mirko Morandini, Loris Penserini, and Anna Perini. Towards goal-oriented development of self-adaptive systems. In *Proceedings of the 2008 international workshop on Software engineering for adaptive and self-managing systems*, pages 9–16, Leipzig, Germany, 2008.
- [OMG01]OMG. Model Driven Architecture, 2001. <http://www.omg.org/mda/>.
- [La03] Axel van Lamsweerde. From system goals to software architecture. In *Formal Methods for Software Architectures*, pages 25–43. Springer-Verlag, 2003.
- [WSB09]Jon Whittle, Pete Sawyer, Nelly Bencomo, Betty H.C. Cheng, and Jean-Michel Bruel. Relax: Incorporating uncertainty into the specification of self-adaptive systems. In *Proceedings of the 2009 17th IEEE International Requirements Engineering Conference, RE*, pages 79–88, Washington, DC, USA, 2009.
- [YLL08] Yijun Yu, Alexei Lapouchnian, Sotirios Liaskos, John Mylopoulos, and Julio Leite. From Goals to High-Variability software design. In *Foundations of Intelligent Systems*, pages 1–16. Springer-Verlag, 2008.
- [ZC06] Ji Zhang and Betty H.C. Cheng. Model-based development of dynamically adaptive software. In *Proceedings of the 28th international conference on Software engineering*, pages 371–380, Shanghai, China, 2006.