Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

Specifying Patterns for Dynamic Pattern Instance Recognition with UML 2.0 Sequence Diagrams

Lothar Wendehals

Universität Paderborn

6. Workshop Software-Reengineering
Bad Honnef, 3. - 5. Mai 2004

Motivation

Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

- Unterstützung des Reverse Engineers durch Design-Rückgewinnung eines Software Systems
- Erzeugung von Design Dokumenten aus Quelltexten
- Dokumentation auf Basis von Klassendiagrammen und Entwurfsmustern

Ziel:

- Werkzeuggestützte Erkennung von Entwurfsmustern
- Formale Spezifikationssprache für Entwurfsmuster

Aktueller Stand

Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

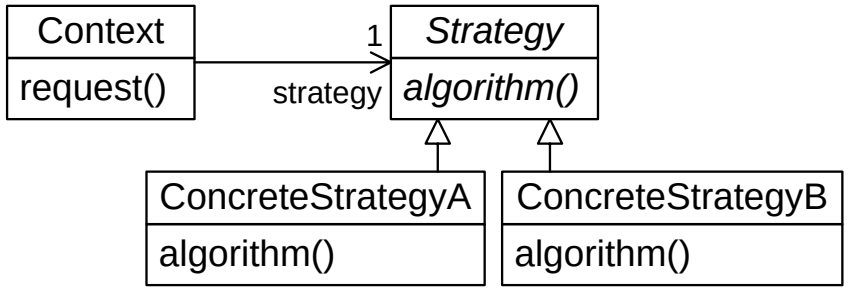
- Statische Analyse auf Abstrakten Syntaxgraphen (ASG)
- Formale, graphische Spezifikation von Entwurfsmustern mit Hilfe von Graphtransformationsregeln
- Spezifikation vor allem von strukturellen Eigenschaften
- Eingeschränkte Analyse der Methodenrümpfe
- Kombination von Teilmustern zu Entwurfsmustern
- Iterative, semi-automatische Analyse
- Unscharfe Bewertung der Analyseergebnisse (Präzisionswerte)

Problem

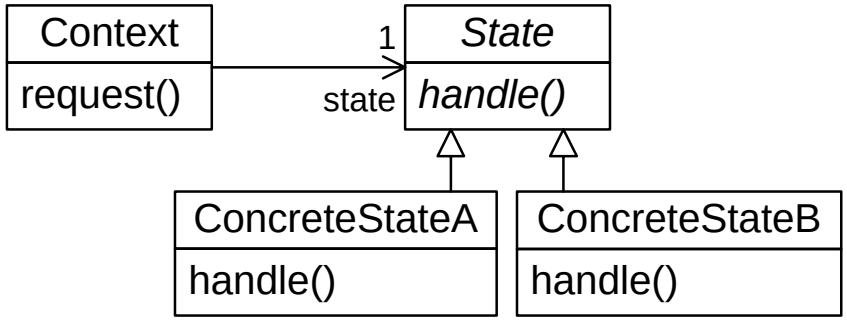
Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

- Strukturell ähnliche Entwurfsmuster statisch nicht unterscheidbar

- Strategy:
  Variiert einen Algorithmus unabhängig vom Kontext, in dem er verwendet wird



- State:
  Ändert das Verhalten eines Objektes abhängig von seinem Zustand



Lösung

Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

- Entwurfsmuster häufig unterscheidbar durch ihr Verhalten
- Verhalten analysierbar durch Methodenaufrufe
- Polymorphie und dynamische Methodenbindung verhindern korrekte statische Analyse
- Statische Analyse gut geeignet für Struktur
- Dynamische Analyse gut geeignet für Verhalten

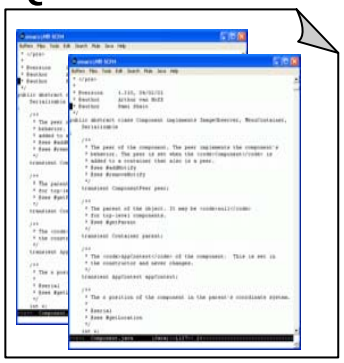
➢ Kombination aus statischer und dynamischer Analyse

➢ Spezifikation von Struktur- und Verhaltensmustern

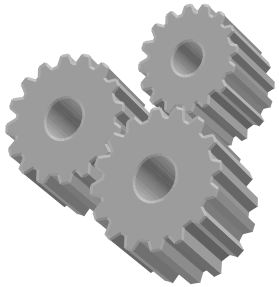
Der Prozess

Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

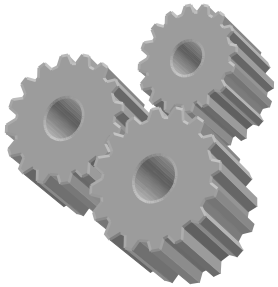
Quelltexte



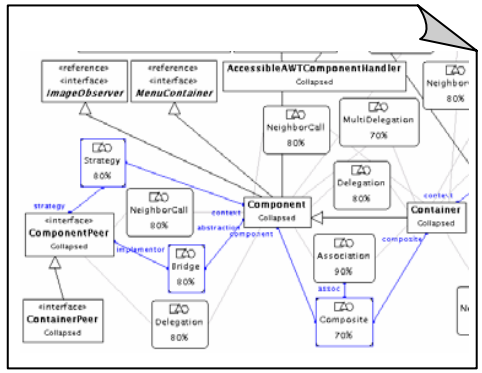
Statische Analyse



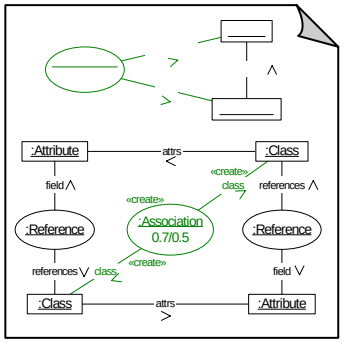
Dynamische Analyse



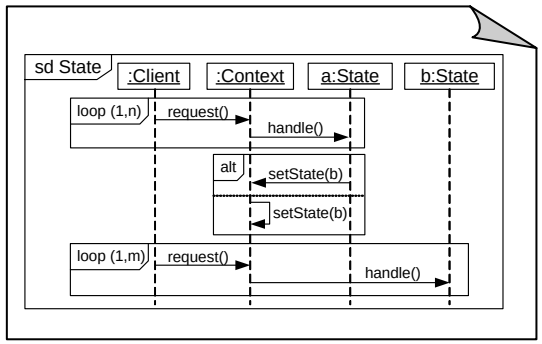
Design Dokumente



Struktur-Muster



Verhaltens-Muster



Tests

Reverse Engineer

Datenfluss

Programm Ausführung

Dynamische Analyse

Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

- Ergebnisse aus statischer Analyse verifizieren
- Beobachtung von Musterinstanz-Kandidaten
- Aufzeichnung eines Aufrufgraphen
- Mustersuche auf dem Aufrufgraphen

➢ Spezifikation von Verhalten?

Spezifikation von Verhaltensmustern

Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

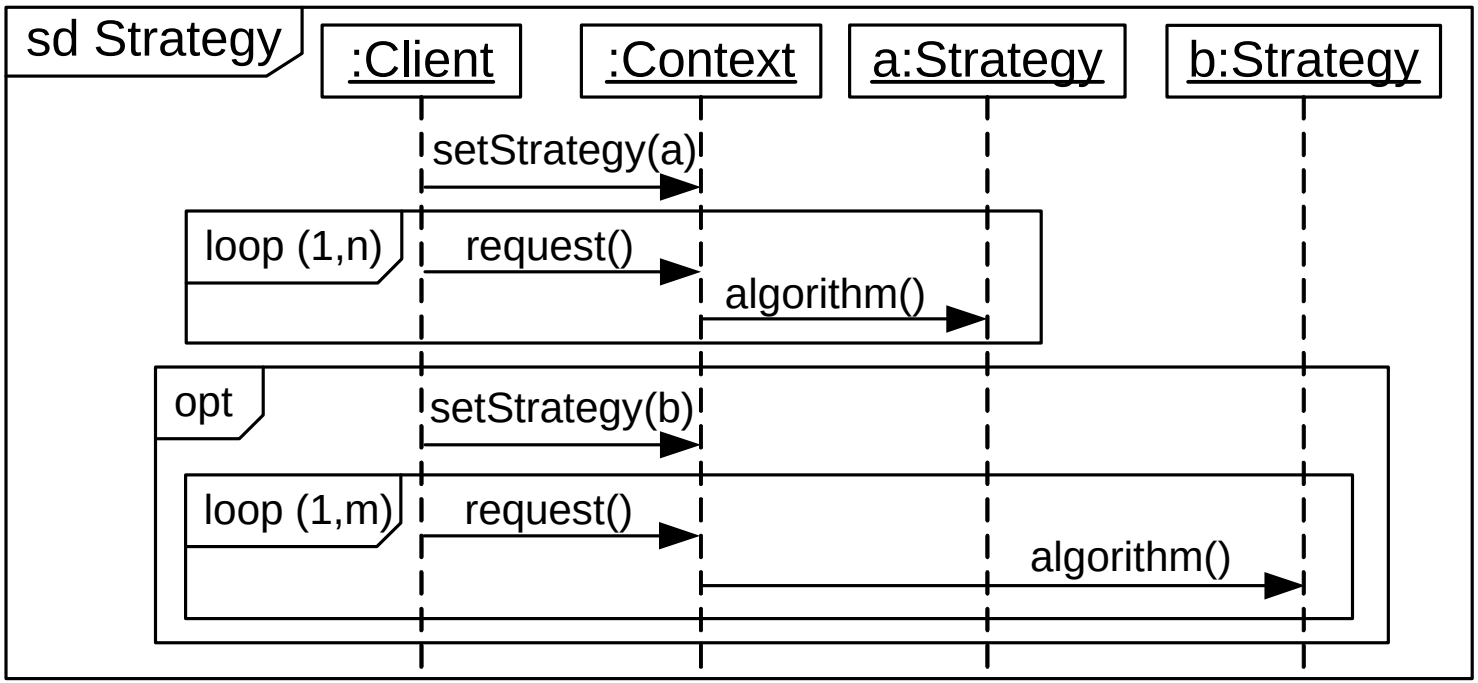
- Übliche Spezifikationssprachen für Muster:
  Prolog, Perl-Skripte, reguläre Ausdrücke,
  Java-Programme
- In statischer Analyse graphische Sprache basierend auf UML-Objektdiagrammen
- In dynamischer Analyse ebenfalls graphische Sprache
- Möglichst intuitiv

➢ Verhalten spezifizieren wie im Forward Engineering:
  UML 2.0 Sequenzdiagramme

Verhaltensmuster - Strategy

Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

- *"A context forwards requests from its clients to its strategy. Clients usually create and pass a ConcreteStrategy object to the context..."**

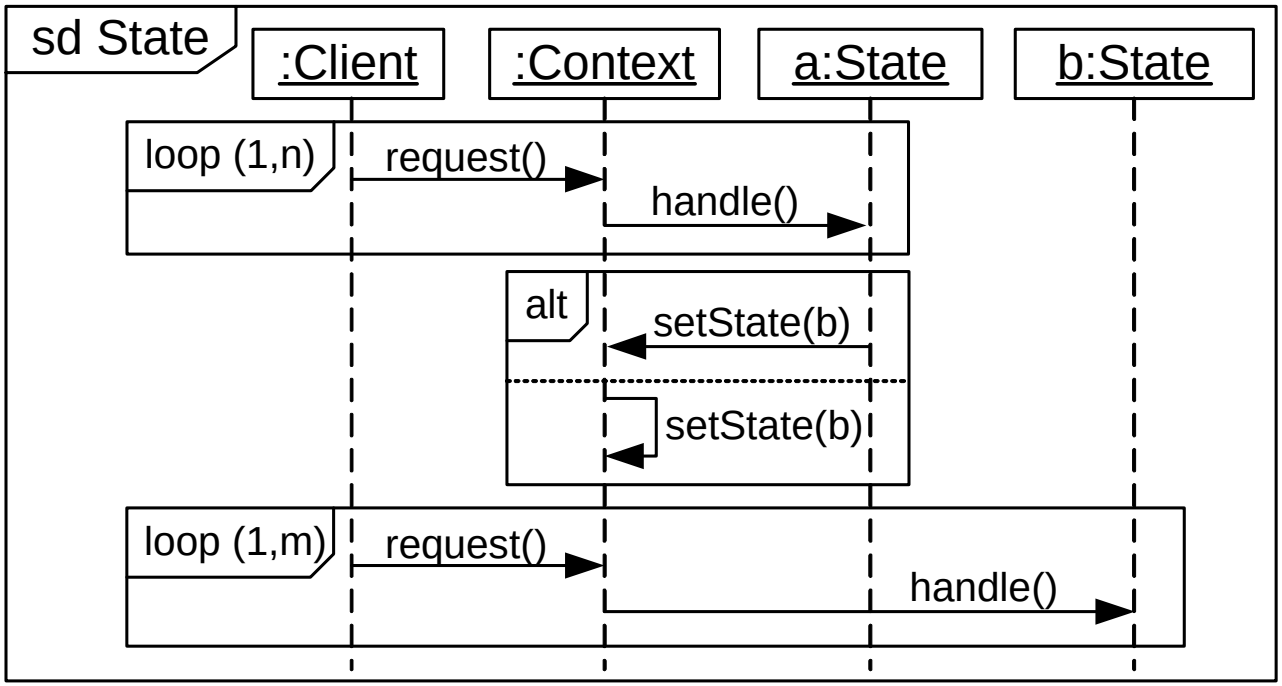


* Gamma, E.; Helm, R.; Johnson, R.; Vlissides J.:
Design Patterns – Elements of Reusable Object-Oriented Software, Addison-Wesley, 1995

Verhaltensmuster - State

Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

- *"... Clients can configure a context with State objects. Once a context is configured, its clients don't have to deal with the State objects directly. Either Context or the ConcreteState subclasses can decide which state succeeds another and under what circumstances."**



* Gamma, E.; Helm, R.; Johnson, R.; Vlissides J.:
Design Patterns – Elements of Reusable Object-Oriented Software, Addison-Wesley, 1995

Semantik der Verhaltensmuster

Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

- Implizite *consider*-Semantik:
  Nur explizit verwendete Methoden werden berücksichtigt
- Verhaltensmuster nur ein „Slice“ aus Aufrufgraphen
- Verhaltensmuster beschreibt Menge von Sequenzen mit gemeinsamer Subsequenz
- Weitere Syntaxelemente:
  - *negative*: Sequenz von Nachrichten darf nicht auftreten
  - *critical region*: Sequenz darf nicht durch andere Nachrichten unterbrochen werden
  - *reference*: Referenziert andere Sequenzen

Aktuelle Arbeiten und Ausblick

Universität Paderborn
AG Softwaretechnik
Prof. Dr. Wilhelm Schäfer

- Komposition von Verhaltensmustern
- Unscharfe Bewertung der Verhaltensmuster-Instanzen
- Verrechnung der Ergebnisse aus statischer und dynamischer Analyse
- Einbindung in die Fujaba Tool Suite

Vielen Dank für Ihre Aufmerksamkeit!

www.FUJABA.de
Tool Suite