

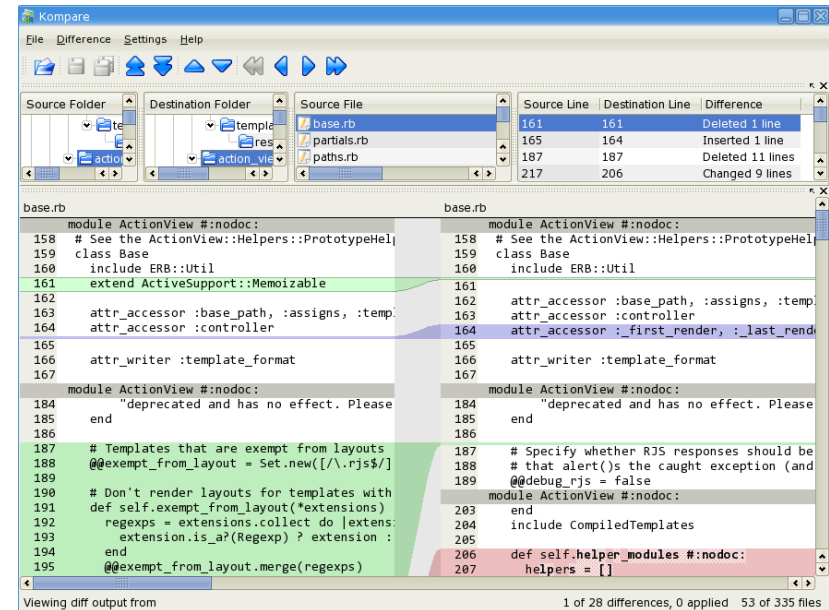
Software Engineering II

Software Engineering II – Übung 6
Wintersemester 12/13
Fachgebiet Software Engineering

Andreas Scharf

Model Diff I

- **Moderne Software besteht aus mehreren Millionen Zeilen Code**
 - Iterativer Entwicklungsprozess → Entwickler Code ändert sich ständig
 - Unterschiede (Deltas) werden in Versionskontrollsystemen verwaltet
- **Frage: Welche Änderungen wurden durchgeführt?**
 - Typisch: Textbasierte Deltaberechnung
 - Anzeige von *gelöschten*, *hinzugefügten* und *geänderten* Zeilen
 - Zusammenführen (Mergen) von Änderungen
 - Funktioniert gut für Textbasierte Dateien, aber...



Source Line	Destination Line	Difference
161	161	Deleted 1 line
165	164	Inserted 1 line
187	187	Deleted 11 lines
217	206	Changed 9 lines

```

base.rb                                     base.rb
-----                                     -----
158  module ActionView #:nodoc:              158  module ActionView #:nodoc:
159  # See the ActionView::Helpers::PrototypeHel| 159  # See the ActionView::Helpers::PrototypeHel|
160  class Base                                159  class Base
161  include ERB::Util                          160  include ERB::Util
162  extend ActiveSupport::Memoizable           161
163  attr_accessor :base_path, :assigns, :temp.  162  attr_accessor :base_path, :assigns, :temp.
164  attr_accessor :controller                  164  attr_accessor :first_render, :last_rend
165  attr_writer :template_format              165
166  attr_writer :template_format              166  attr_writer :template_format
167  attr_writer :template_format              167
168  attr_writer :template_format              168
184  module ActionView #:nodoc:                184  module ActionView #:nodoc:
185  "deprecated and has no effect. Please     185  "deprecated and has no effect. Please
186  end                                         186  end
187  # Templates that are exempt from layouts   187  # Specify whether RJS responses should be
188  @@exempt_from_layout = Set.new([/\.rjs$/]  188  # that alert()s the caught exception (and
189  @debug_rjs = false                         189  @debug_rjs = false
190  # Don't render layouts for templates with  203  end
191  def self.exempt_from_layout(*extensions)  204  include CompiledTemplates
192  regexprs = extensions.collect do |extens:  205
193  extension.is_a?(Regexp) ? extension :    206  def self.helper_modules #:nodoc:
194  end                                         207  helpers = []
195  @@exempt_from_layout.merge(regexprs)     207

```

Quelle: [1]

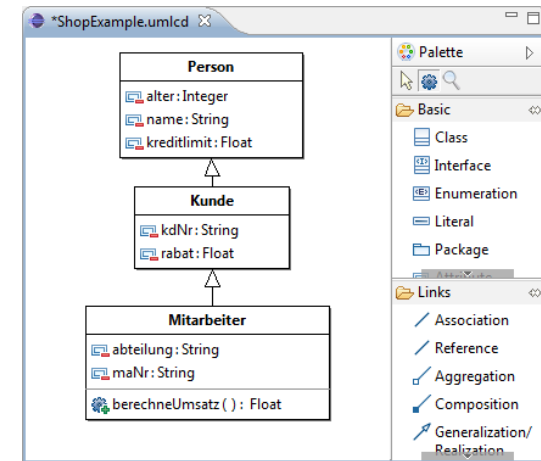
Model Diff II

- **Planung von Struktur und Verhalten hat sich etabliert**

- Häufig UML (z.B. Klassendiagramme)
- Diagramme visualisieren Modelle

➔ **Modelle vs. Text**

- Spezielle Editoren zum Anzeigen und Bearbeiten

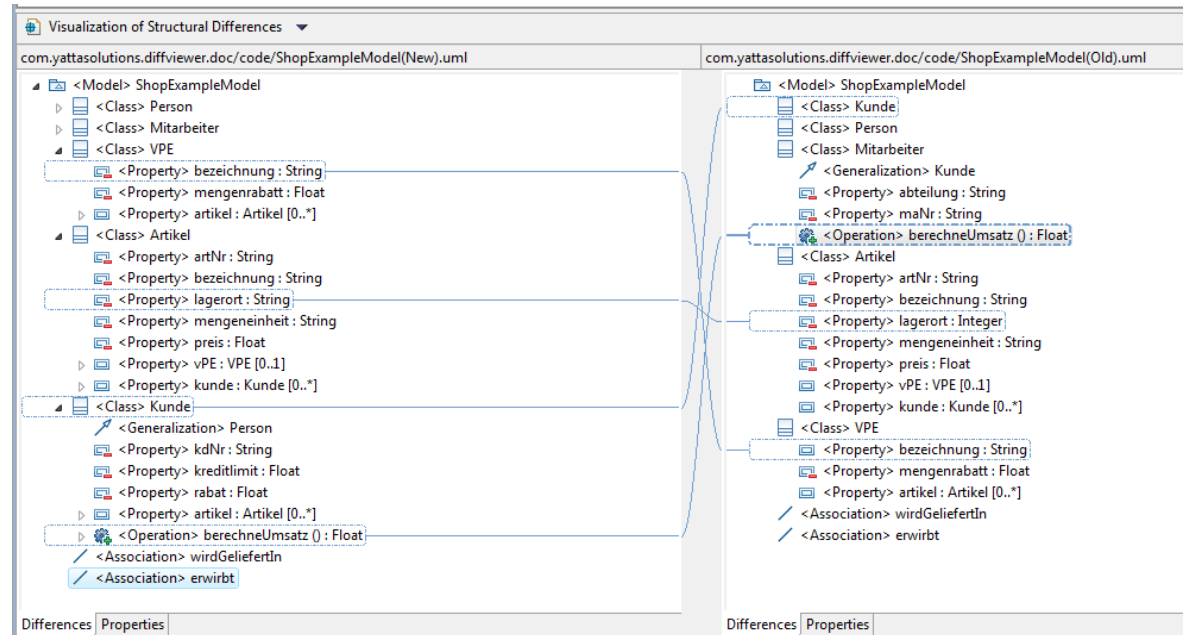


- **Typische Algorithmen zur Deltaberechnung und Anzeige nicht anwendbar**

- Speichern als XMI hilft nicht:
 - Verschieben äußert sich in Entfernen und Hinzufügen
 - Reihenfolge von Elementen im Modell u.U. nicht wichtig
 - Etc.

Model Diff III

- Spezielle Algorithmen zum Modellvergleich
- Verschiedene Darstellungsmöglichkeiten:
 - Generische Baumdarstellung
 - ✘ Unübersichtlich
 - ✘ Ungewohnte Darstellung



Quelle: [2]

Model Diff IV

- **Änderungen sollen direkt im Diagramm angezeigt werden**
 - Gewohnte Darstellung
 - Generische Visualisierung
- **Integration in bestehende Editoren**
 - Entwicklung ist komplex und kostenintensiv
 - Integrationsaufwand minimieren
- **Mergen direkt im Diagramm**
- **Vergleich lokaler Modelle und gegen VCS**

Literatur

- [1] KEEL, Joshua ; SNYDER, Jeff: Kompare - Different from the rest. 2005. – URL <http://www.caffeinated.me.uk/kompare/>
- [2] SKRYPUCH, Neil: EMF Compare. – URL <http://www.eclipse.org/modeling/emf/?project=compare>