

# Refactoring in der Praxis

- Einleitung
- Was kann Refactoring bewirken?
- Welche Refactoring-Tools gibt es?
- Ausgewählte Refactoring-Methoden
- Wie kann man seinen Erfolg durch Refactoring meßbar machen?

# Einleitung

- „Beim Refactoring wird der Quelltext eines Computerprogramms umgestaltet, wobei die tatsächliche Programmfunktion unverändert bleiben soll“
- Begriff „Code Smell“
- Erstmals 1990 erwähnt von Ralph Johnson (GoF) und William Odyke
- Bekannt geworden durch „Refactoring. Wie Sie das Design vorhandener Software verbessern“ von Martin Fowler. (siehe auch Seite [www.refactoring.com](http://www.refactoring.com))
- Es gibt über 90 verschiedene Refactorings (<http://www.refactoring.com/catalog/index.html>)

# Literaturhinweise

- Martin Fowler:  
*Refactoring. Wie Sie das Design vorhandener Software verbessern*, Addison-Wesley Verlag,
- William C. Wake:  
*Refactoring Workbook*, Addison-Wesley,  
ISBN 0-321-10929-5
- Joshua Kerievesky:  
*Refactoring To Patterns*, Addison-Wesley,  
ISBN 0321213351

# Refactoring in der Praxis

- Einleitung
- Was kann Refactoring bewirken?
- Welche Refactoring-Tools gibt es?
- Ausgewählte Refactoring-Methoden
- Wie kann man seinen Erfolg durch Refactoring meßbar machen?

# Was kann Refactoring bewirken?

- Lesbarkeit
- Übersichtlichkeit
- Verständlichkeit
- Erweiterbarkeit
- Vermeidung von Redundanz
- Testbarkeit

# Übersichtlichkeit

- Praxisbeispiel:  
Entfernung von 9000 Codezeilen durch Refactoring bei gesteigerter Funktionalität
- Reduzierung der Methode „BindMask“ von 491 auf 98 Codezeilen.

# Refactoring in der Praxis

- Einleitung
- Was kann Refactoring bewirken?
- ***Welche Refactoring-Tools gibt es?***
- Ausgewählte Refactoring-Methoden
- Wie kann man seinen Erfolg durch Refactoring meßbar machen?

# Welche Tools gibt es für C#

- C# Refactory  
<http://www.xtreme-simplicity.net/CSharpRefactory.html>
- ReSharper  
<http://www.jetbrains.com/resharper/>
- devAdvantage  
<http://www.anticipatingminds.com/>
- C# Refactoring Tool  
<http://dotnetrefactoring.com/>
- .NET Refactor  
<http://www.hhisoftware.com/>

# Tools für andere Sprachen

- C++
  - Xrefactory  
<http://xref-tech.com/xrefactory/main.html>
  - Ref++ (Addin für Visual Studio)  
<http://www.ideat-solutions.com/refpp/>
- Java
  - IntelliJ (IDE mit über 35 Refactoringmethoden)  
<http://www.jetbrains.com>
- VB
  - Refactor! Pro  
<http://www.devexpress.com/Products/NET/Refactor/>

# Refactoring in der Praxis

- Einleitung
- Was kann Refactoring bewirken?
- Welche Refactoring-Tools gibt es?
- ***Ausgewählte Refactoring-Methoden***
- Wie kann man seinen Erfolg durch Refactoring meßbar machen?

# Ausgewählte Refactoringmethoden

- Extract Method
- Introduce Variable
- Extract Interface
- Property too Method und umgekehrt
- Encapsulate Field
- Change Signature

# Refactoring in der Praxis

- Einleitung
- Was kann Refactoring bewirken?
- Welche Refactoring-Tools gibt es?
- Ausgewählte Refactoring-Methoden
- ***Wie kann man seinen Erfolg durch Refactoring meßbar machen?***

# Wie kann man seinen Erfolg durch Refactoringmeßbar machen?

- DevMetrics
  - Verringerung der Codezeilen
  - Sichtbarmachung der geringeren Komplexität
- Simian
  - Doppelte Codezeilen sichtbar machen
- Ohne Softwareunterstützung
  - Zeitliche Darstellung von Umstellungsprojekten ohne und mit Refactoring
  - Realisierung von Projekten, die ohne Refactoring nicht möglich wären.

**Ende**

**Vielen Dank für Ihre Aufmerksamkeit!**