



Auf der Suche nach Werten in der Softwaretechnik:

Werte und Objekte in objektorientierten Programmiersprachen

Jörg Rathlev

Beate Ritterbach

Axel Schmolitzky

29. März 2007



DAS Grundprinzip der Objektorientierung:

„Alles ist ein Objekt“

Wirklich alles?



Die Frage ist nicht:

Werte - ja oder nein?

Sondern:

Werte - explizit oder implizit?

Was sind Werte?

- Zielsetzung
 - ◆ Verhalten wie primitive Typen
 - ◆ aber vom Programmierer definierbar
 - Unerzeugbar
 - Unveränderbar
 - Seiteneffektfrei und referentiell transparent
 - Dies sind konzeptionelle Eigenschaften!
-
-

Vorsicht – Verwechslungsgefahr!

- „Wert“ einer Variablen (besser: Belegung)
- „Wertsemantik“ (vs. Referenzsemantik)
- „Übergabe per Wert“ (vs. „per Referenz“)



Unterstützung in oo Sprachen für...

	Wert-Semantik	Referenz-Parameter	Benutzer-definierte Wert-Typen
Smalltalk	–	–	–
Eiffel	+	–	–
C++	+	+	–
Java	–	–	–
C#	+	+	–

Werttypen mit Objekttypen vortäuschen

- Unveränderlichkeit ist machbar
- Unerzeugbarkeit ist eher schwierig...

- Umsetzung klassisch durch
 - ◆ Frameworks
 - ◆ Entwurfsmuster
 - ◆ Konventionen



Sprach-Unterstützung für benutzerdefinierte Werttypen

Vorteile:

- Klarheit
 - ◆ Werttypen im Quelltext als solche erkennbar
 - ◆ Prägnantere Quelltexte
- Sicherheit
 - ◆ Eigenschaften von Werten sind garantiert, vom Compiler überprüft, beruhen nicht auf Konventionen
 - ◆ „Wertartigkeit“ kann nicht versehentlich durch Überarbeitung zerstört werden

Beispiel-Implementierung: VJ

- Erweiterung der Sprache Java
- Unterstützung für strukturierte Werttypen
 - ◆ „konstituierende Eigenschaften“
 - ◆ Typdefinition mit „valueclass“
- Werteigenschaften sprachseitig garantiert
- VJ-Compiler:
 - ◆ <https://sourceforge.net/projects/vj-lang>