

Wert (Programmierung)

Wechseln zu:[Navigation](#), [Suche](#)

Dieser Artikel erfüllt die [GlossarWiki-Qualitätsanforderungen](#) nur teilweise:

Korrektheit: 5 (vollständig überprüft)	Umfang: 4 (unwichtige Fakten fehlen)	Quellenangaben : 2 (wichtige Quellen fehlen)	Quellenarten: 3 (gut)	Konformität: 5 (ausgezeichnet)
--------------------------------------------------	------------------------------------------------	-----------------------------------------------------------	---------------------------------	------------------------------------------

Inhaltsverzeichnis

- 1 Definition (für nicht-objektorientierte Programmierparadigmen)
- 2 Definition (für objektorientierten Programmierparadigmen)
- 3 Bemerkungen
 - 3.1 Bedeutung von „systemweit“
 - 3.2 Datentyp
- 4 Beispiele
- 5 Quellen
- 6 Siehe auch

1 Definition (für nicht-objektorientierte Programmierparadigmen)

Ein **Wert** ist (im Sinne der meisten **Programmierparadigmen**) ein **Datum** (Einzahl von Daten), das folgende Merkmale aufweist:

Ein Wert existiert dauerhaft, d.h., er wird weder erzeugt, noch kann er zerstört oder verändert werden.

Ein Wert w hat einen unveränderlichen, **systemweit** eindeutigen und nicht anderweitig verwendbaren **Wertidentifikator** (WID) $wid(w)$, mit dem er jederzeit eindeutig identifiziert werden kann.

Neben dem Identifikator kann es zu jedem Zeitpunkt t für einen Wert w noch einen oder mehrere **Wertnamen** $wnamen(w,t)$ geben. Diese sind im Gegensatz zum WID i.Allg. veränderlich, wiederverwendbar und nicht notwendigerweise systemweit eindeutig. Mit Hilfe eines Namens kann ein Wert daher nur unter bestimmten Bedingungen zu bestimmten Zeiten eindeutig identifiziert werden.

Für jeden Wert kann es eine Menge von **Anfragemethoden** geben, mit deren Hilfe spezielle Eigenschaften des Wertes ermittelt werden können.

2 Definition (für objektorientierten

Programmierparadigmen)

Ein **Wert** ist im Sinne der **Objektorientierten Programmierung** ein **Objekt**, das folgende Merkmale aufweist:

Ein Wert existiert dauerhaft, d. h., er wurde „zu Anbeginn der Zeit“ erzeugt und wird erst „zum Ende der Zeit“ zerstört. Es gibt also keine **Konstruktoren** und **Destruktoren** für Werte.

Der Zustand eines Wertes kann nicht verändert werden.

Der **Objektidentifikator** (OID) eines Wertes w wird i. Allg. **Wertidentifikator** (WID) genannt: $wid(w) := oid(w)$.

Ein **Objektname** eines Wertes wird i. Allg. **Wertname** genannt.

Jeder Wert hat zu jedem Zeitpunkt mindestens einen Wertnamen.

Für jeden Wert kann es eine Menge von **Anfragemethoden** geben, mit deren Hilfe spezielle Eigenschaften des Wertes ermittelt werden können. **Modifikationsmethoden** gibt es für Werte jedoch nicht.

3 Bemerkungen

3.1 Bedeutung von „systemweit“

Der in der Definition verwendete Begriff „*systemweit*“ ist nicht eindeutig definiert, sondern hängt von der jeweiligen Systemumgebung ab. Bei einem einfachen Programmlauf eines **JAVA**-Programms beispielsweise bedeutet „*systemweit*“ für ein Wert „während der Lebensdauer des Programms (die endet spätestens mit dem Programmende) innerhalb der **Laufzeitumgebung** des Programms“. Für ein verteiltes objektorientiertes Datenbanksystem bedeutet „*systemweit*“ dagegen „über alle beteiligten Datenbanken hinweg solange die zugehörigen Datenbanken existieren“.

3.2 Datentyp

Der Begriff **Datentyp** wird meist in einem Atemzug mit dem Begriff **Wert** genannt. Ein Datentyp beschreibt eine **Menge** von Werten, für die bestimmte **Funktionen** definiert sind.

Man beachte, dass die obige Definition ohne den Begriff **Datentyp** auskommt.

4 Beispiele

Die **natürliche Zahl** 75 ist ein typischer Wert, genauer gesagt, ein typischer **Wertname**. Mögliche **Wertnamen**, die alle denselben Wert bezeichnen, sind:

75

+75

075 (wenn führende Nullen ignoriert werden) oder 0113 (wenn eine führende Null ausdrückt, dass die Zahl in **oktaler** Schreibweise dargestellt wird)

0x4b (**hexagesimale Darstellung**)

etc.

Der zugehörige **Wertidentifikator** hängt vom verwendeten Programmiersystem ab.

Beispiele:

8-stellige **Dualzahl** 0100 1011 (Datentyp **Tiny Integer**)

16-stellige **Dualzahl** 0000 0000 0100 1011 (Datentyp **Small Integer**)

32-stellige **Dualzahl** 0000 0000 000 0000 000 0000 0100 1011 (Datentyp **Integer**)

etc.

Ein weiteres Beispiel ist der Buchstabe K. Auch dieser Wert hat verschiedene Wertnamen:

K

\x04b

\113

etc.

Der zugehörige **Wertidentifikator** hängt wiederum vom verwendeten Programmiersystem ab.

Beispiele:

8-stellige **Dualzahl** 0100 1011 (**ASCII**-Code)

16-stellige **Dualzahl** 0000 0000 0100 1011 (**UTF-16**-Code)

8-stellige **Dualzahl** 1101 0010 (**EBCDIC**-1148-Code)

etc.

Es fällt auf, dass die möglichen Wertidentifikatoren dieser beiden Werte teilweise übereinstimmen. Zum Beispiel ist der Wertidentifikator der Small-Integer-Zahl 75 gleich dem **UTF-16**-Code des Buchstaben K. Dies widerspricht jedoch der Forderung nach „systemweiter Eindeutigkeit“ des Wertidentifikators.

Viele Programmiersysteme erzwingen diese Eindeutigkeit, indem sie z.B. jeden Wertidentifikator um einen eindeutigen Typidentifikator erweitern. Es gibt allerdings auch Sprachen, wie z.B. C++, die aus Performanzgründen die systemweite Eindeutigkeit von Wertidentifikatoren nicht automatisch erzwingen. Das hat zur Folge, dass zur Laufzeit eines Programms zwei unterschiedliche Werte in bestimmten Situationen nicht mehr unterschieden werden können (z.B. nachdem diese Werte in eine **Liste** oder in ein **Array** eingefügt wurden).

5 Quellen

Kowarschick (MMProg): Wolfgang Kowarschick; Vorlesung „Multimedia-Programmierung“; Hochschule: [Hochschule Augsburg](#); Adresse: [Augsburg](#); [Web-Link](#); 2018; **Quellengüte**: 3 (Vorlesung)

Kowarschick (2002a): Wolfgang Kowarschick; Multimedia-Programmierung – Objektorientierte Grundlagen; Hrsg.: Michael Lutz und Christian Märtin; Reihe: [Informatik interaktiv](#); Verlag: [Fachbuchverlag Leipzig im Carl Hanser Verlag](#); ISBN: 3446217002; 2002; **Quellengüte**: 5 (Buch)

Kowarschick (2002): Wolfgang Kowarschick; Vorlesung „Multimedia Softwareentwicklung II“ – Wintersemester 2001/2002; Hochschule: [Fachhochschule Augsburg](#); Adresse: [Augsburg](#); [Web-Link](#); 2002; **Quellengüte**: 4 (Skript)

6 Siehe auch

[Objekt](#)

[WikipediaEN:Value \(computer science\)](#)

Kategorien:

Datentyp

Objektorientierte Programmierung

Glossar

Kapitel:Multimedia-Programmierung

Diese Seite wurde zuletzt am 3. August 2019 um 15:08 Uhr bearbeitet.

Inhalt verfügbar unter [CC BY-SA 4.0](#).

