Universität Zürich[UZH]

Institut für Informatik

Martin Glinz

Requirements Engineering I

Kapitel 9

UML

Unified Modeling Language

© 2006-2011 Martin Glinz. Alle Rechte vorbehalten. Speicherung und Wiedergabe für den persönlichen, nicht kommerziellen Gebrauch gestattet; bei auszugsweiser Verwendung mit Quellenangabe. Verwendung für Unterrichtszwecke oder kommerziellen Gebrauch nur mit vorheriger schriftlicher Genehmigung des Autors.

9.1 Grundlagen und Definitionen

- UML (Unified Modeling Language) ist eine Sammlung vorwiegend grafischer Sprachen zur Erstellung von Anforderungs- und Entwurfsmodellen aus verschiedenen Perspektiven
- Keine kohärente Modellierungssprache
- Entstanden aus der Absicht, einen de-facto Industriestandard zu schaffen
- Standardisiert durch die OMG (Object Management Group; ein Industriekonsortium)
- Heute die dominierende Modellierungssprache

9.2 Geschichte von UML

- Zwischen 1990 und 1995 entsteht eine Vielzahl von Techniken und Sprachen zur objektorientierten Modellierung von Entwürfen und Anforderungen für Software
- 1994/95 tun sich drei führende Exponenten der damaligen Ansätze (Grady Booch, Jim Rumbaugh, Ivar Jacobson) zusammen, um eine "Unified Method" als Industriestandard zu schaffen
- Der Versuch einer Einheitsmethode scheitert. Statt dessen wird 1996 der Entwurf einer einheitlichen Sprache, der "Unified Modeling Language (UML)" publiziert.
- Mehrere große Firmen (IBM, Oracle, HP,...) beteiligen sich und bringen Ausbauwünsche ein: UML wächst
- 1997 wird UML 1.1 von der OMG zu ihrer Standard-Modellierungs-sprache bestimmt

Geschichte von UML – 2

- UML setzt sich rasch als Standardsprache für die grafische Modellierung von Anforderungen und Entwürfen durch
- Gleichzeitig erfährt UML eine Reihe von Revisionen:
  - 1997: UML 1.0
  - 1997: UML 1.1 — von OMG verabschiedete Version
  - 1998: UML 1.2
  - 1999: UML 1.3
  - 2001: UML 1.4
  - 2003: UML 1.5 — Vorgriff auf 2.0; nie in der Breite eingesetzt
  - 2004: UML 2.0 — erhebliche Änderungen
  - 2007: UML 2.1
  - 2009: UML 2.2
  - 2010: UML 2.3

  } Fehlerkorrekturen und geringfügige Anpassungen
- Heute in Gebrauch: 2.0 bis 2.3, stellenweise noch 1.4

9.3 Grundkonzepte der UML

- Eine UML-Spezifikation besteht aus einer Sammlung sich ergänzender und teilweise überlappender Modelle
- Im Zentrum steht ein Klassenmodell, das den strukturellen Aufbau eines Systems spezifiziert
- Nach Bedarf beschreiben weitere Modelle zusätzliche Sichten, insbesondere dynamisches Verhalten, Funktionalität und Zusammenarbeit
- Alle grafischen Konstrukte können mit Text annotiert oder mit der formalen Spezifikationssprache OCL zusätzlich eingeschränkt werden

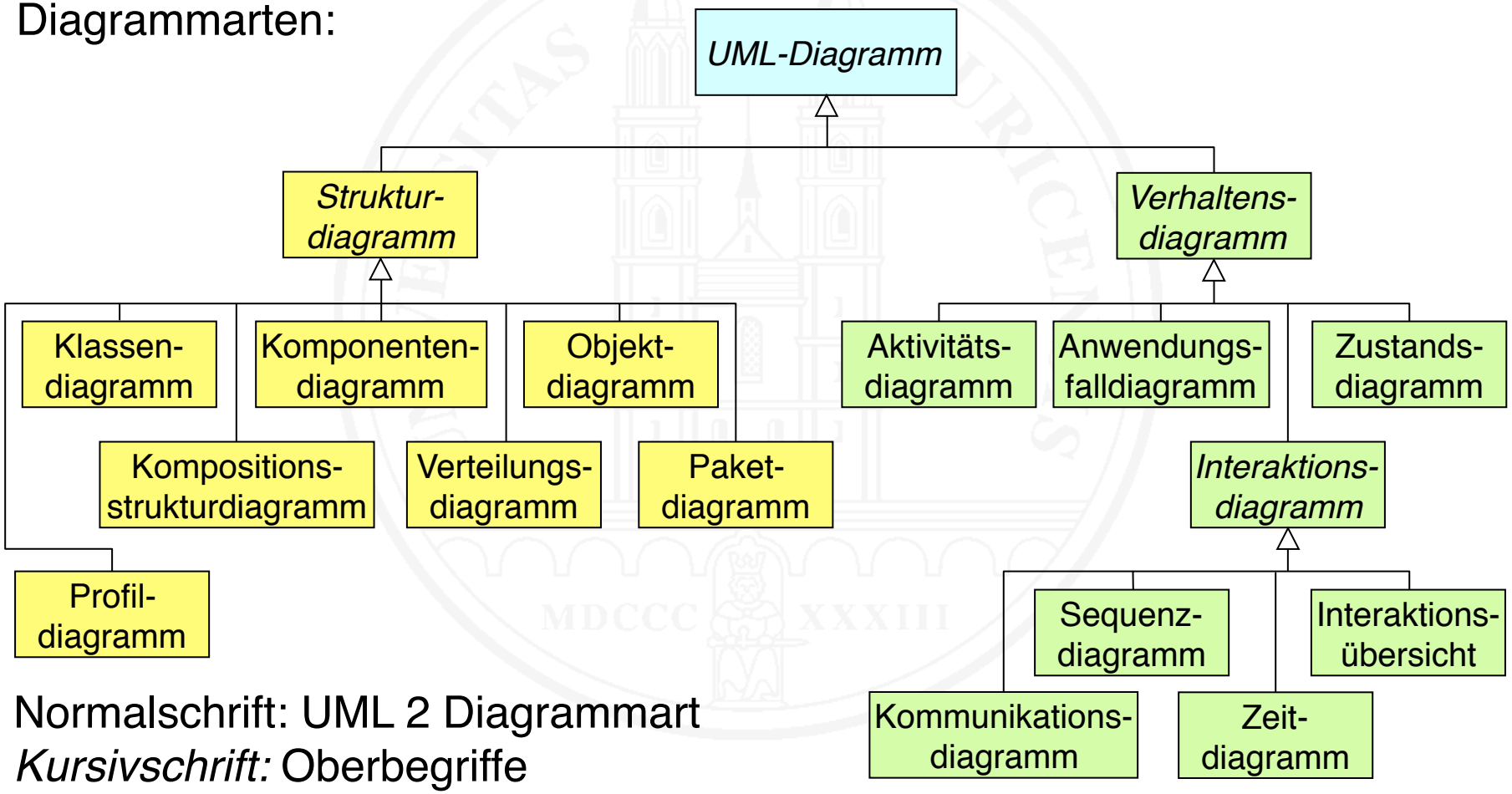
Modellierbare Aspekte in UML

UML unterstützt insgesamt acht verschiedene Sichten:

- Strukturelle Sichten
  - Statische Struktur (Daten-, Klassen und Objektmodelle)
  - Struktur und Zusammenarbeit von Komponenten (ab UML 2)
  - Portionierung von UML-Modellen in Pakete und Subsysteme
  - Physische Struktur: Artefakte, Knoten, Verteilung
- Verhaltenssichten
  - Interaktion externer Akteure mit einem System
  - Zeitlich-dynamisches Verhalten von Systemen
  - Interaktion ausgewählter Objekte untereinander
  - Aktivitäten und deren Ablauf

Die UML-Diagrammarten (UML 2.3)

Zur Darstellung der genannten Sichten verwendet UML 2 die folgenden Diagrammarten:



Normalschrift: UML 2 Diagrammart
Kursivschrift: Oberbegriffe

9.4 Anforderungsspezifikation mit UML

- Gliederung einer Aufgabe in Teilaufgaben
  - Kompositionsstrukturdiagramme oder Paketdiagramme
- Einbettung eines Systems in eine Systemlandschaft
  - Komponentendiagramme
- Objekte und Operationen
  - Klassendiagramm
- Szenarien und Anwendungsfälle
  - Übersicht: Anwendungsfalldiagramm
  - Einzelne Anwendungsfälle:
    - Strukturierter Text: nicht unterstützt
    - Teilformal: Zustands-, Aktivitäts-, Sequenzdiagramme
  - Zerlegung und Zusammenhänge: nicht unterstützt

Anforderungsspezifikation mit UML – 2

- Prozesse
  - Aktivitätsdiagramme
- Dynamisches Systemverhalten
  - Zustandsdiagramme, sekundär Sequenz- oder Aktivitätsdiagramme
- Hauptprobleme:
  - Konsistenz der Teilmodelle untereinander
  - Herstellung systematischer Zusammenhänge zwischen Teilmodellen
  - Zusammensuchen aller relevanten Anforderungen zu einem Teilproblem

9.5 UML – Bewertung

+ Standardisiert und weit verbreitet
+ Modellierungsmittel für die meisten Probleme im Software Engineering
+ Breites Angebot an Werkzeugen, Büchern und Schulung

– (Zu) großer Sprachumfang
– Konzeptionelle Schwächen, im Requirements Engineering insbesondere bei der Formulierung von Anwendungsfällen
– UML-Modelle sind Sammlungen von Einzelmodellen: Konsistenzprobleme, Problem des Zusammensuchens relevanter Information

Literatur

Fowler, M. (2003). *UML Distilled: A Brief Guide to the Standard Object Modeling Language*. 3rd Edition. Boston: Addison-Wesley.

Glinz, M. (2000). Problems and Deficiencies of UML as a Requirements Specification Language. *Proceedings of the Tenth International Workshop on Software Specification and Design*. San Diego. 11-22.

Oestereich, B. (2004). *Objektorientierte Softwareentwicklung: Analyse und Design mit der UML 2.0*. 6. Auflage. München: Oldenbourg Verlag.

Object Management Group (2010). *Unified Modeling Language: Superstructure,* version 2.3. OMG document formal/2010-05-05. http://www.omg.org/spec/UML/2.3/Superstructure/PDF

Rumbaugh, J., Jacobson, I., Booch, G. (1999). *The Unified Modeling Language Reference Manual*. Reading, Mass. : Addison-Wesley.