

Software-Entwicklungswerkzeuge (SEW)

1

Prof. Dr. Uwe Aßmann
Technische Universität Dresden
Institut für Software- und
Multimediatechnik
<http://st.inf.tu-dresden.de>
WS 12/13-1.0, 10/6/12

- Contents
- Goals

Grobgliederung der SEW-Vorlesung

2

Teil I: Grundlagen

- ▶ Einführung zu Werkzeugen, Werkzeugmaschinen und Software-Entwicklungsumgebungen (SEU)
- ▶ Basistechniken in Werkzeugen
- ▶ Sprachfamilien in Werkzeugen

Teil II: Architektur von Werkzeugen

- ▶ Architektur von SEU
- ▶ Repository
- ▶ Komposition

Teil III: Code-zentrierte Werkzeuge zur Programmanalyse

- ▶ Parser-Generatoren
- ▶ Werkzeuge zur Programmanalyse

Teil VI: Werkzeuge zur Programmtransformation

- ▶ Werkzeuge zur Transformation
- ▶ Werkzeuge zur Optimierung
- ▶ Werkzeuge zur Programmüberführung (Codegenerierung)
- ▶ Dokumentationswerkzeuge
- ▶ MDA-Werkzeuge

Teil V: Werkzeuge im Software-Lebenszyklus

- ▶ Werkzeuge zur Anforderungsanalyse
- ▶ Testwerkzeuge
- ▶ Werkzeuge für Baumanagement
- ▶ Werkzeuge für Wartung und Reengineering
- ▶ Erfahrungen und weitere Entwicklungen

Teil VI: Fortgeschrittene Themen

Gliederung Teil I - Grundlagen

3

Teil I: Grundlagen

- ▶ 10. Taxonomie von Werkzeugen u. Software-Entwicklungsumgebungen (SEU)
 - 1 Aufbau u. prinzipielle Funktion von Software-Entwicklungswerkzeugen
 - 2 Werkzeuggrundtypen - Klassen von CASE-Tools
 - 3 Werkzeug-Landschaft nach Hesse
 - 4 Einführung in Effektkategorien (Blutgruppen)
 - 5 Graph-Logik-Isomorphismus
- ▶ 11. Basistechniken: Metamodellierung und Technologieräume
 - Lifting of M2 models
- ▶ 12. Sprachfamilien in Werkzeugen
 - 1 Begriffsbestimmung
 - 2 DDL: RM, ERD als isomorphes RM, EBNF, CD
 - 3 DQL: Xcerpt, Semmle .QL
 - 4 DCL: OCL, Spider, URML
 - 5 DML: Term- und Graphersetzung
 - 6. DRL: Data Restructuring Languages
 - 7 BSL: DFD in SA, Mashups
- 13 Interpretation and Semantics
- 14 Abstract interpretation of BSL
- 15 Behavioral model checking of automata-based BSL
 - 1 Standard model checking
 - 2 Software model checking
 - 3 Real-time model checking
- Optional: Wiederholungen von ST-1 und ST-2
 - 19.1 Entity-Relationship-Diagramme
 - 19.2 Data Dictionary
 - 19.3 Datenflussdiagramme
 - 19.4 Minispezifikationen (Pseudocode)
 - 19.5 Zustandsgraphen
 - 19.6 Entscheidungstabellen
 - 19.7 Objektorientierte Diagramme

Gliederung Teil II – Architektur von Werkzeugen

4

Teil II: Architektur von Werkzeugen

▶ 20. Architektur von Software-Entwicklungsumgebungen (SEU)

- 1 Konzepte der Werkzeugintegration
- 2 Architektur von SEU
- 3 Das ECMA Referenzmodell

▶ 21. Repository

- 1 Ziele und Aufgaben des Repository
- 2 Architektur von Repositories
- 3 Syntax-Mapping und CASE-Austauschformate
- 4 Frameworks zur Werkzeugintegration (PCTE)

▶ 22. Komposition von repository-basierten Werkzeugen

- 1 Effektkategorien und Blutgruppen
- 2 Werkzeuge als Objekte
- 3 Einsatz des Graph-Logik-Isomorphismus
- 4 Essentielle Zerlegung (EAI)
- 5 Repository-Integration mit Rollen

▶ 23. Komposition von Werkzeugen

- Datenfluss-basierte Integration

▶ (23b Prozessintegration)

- Prozessmodelle und ihre Werkzeuge
- SuReal-Umgebung
- Metamodellierung von Prozessen

24 Syntaxmapping mit EMFText

- Für Syntaxmapping, Technologieraumbrücken und DSL-Entwicklung

▶ 25. MetaCase-Werkzeuge

- Metamodellierung
- MetaCASE-Umgebungen
- MOFLON
- FlowR from Achievo

▶ 27 Werkzeuge zur Konstruktion domänenspezifischer Sprachen

- (EMFText)
- Reuseware zur Komposition von DSL-Fragmenten

▶ (28 Technologieraumbrücken)

- Ontologyware---Modelware
- OWLText

Gliederung Teil III+IV

5

Teil III: Code-zentrierte Werkzeuge zur Programmanalyse

- ▶ 30. Technologieraum Grammarware
 - Moderne Parsergeneratoren
 - 30.1 ANTLR
- ▶ 32 Visualisierung
 - VCG, AiSee, LogLin und CroCoCosmos
- ▶ 33 Werkzeuge zur Programmanalyse mit Graphersetzung
 - Control-Flow Analysis with EARS
- ▶ 34 Interprocedural Analysis with PAG
- ▶ 35 WCETA mit AiT
- ▶ 36 Interprocedural Analysis with IDFS

Teil IV: Werkzeuge zur Code-Transformation

- ▶ 40 Fujaba
 - 34.1 Graphersetzung
- ▶ 45. Werkzeuge zur Codegenerierung (Programmüberführung)
 - Codeumwandlung
 - Codeergänzung und -selektion
 - Codemodifikation und -rückführung
- ▶ 46 Dokumentationswerkzeuge
- ▶ 47 Werkzeuge für die MDA
- ▶ (48 Werkzeuge zur Termersetzung)
 - Stratego

Gliederung Teil V+VI

6

Teil V: Werkzeuge im Software-Lebenszyklus

- ▶ 50. Werkzeuge zur Anforderungsanalyse
 - Werkzeuggestützte Anforderungsanalyse
 - Prüf-, Dokumentations- und Verwaltungswerkzeuge
 - **Geschäftsregelanalyse (RuleXpress)**
- ▶ 51. Testwerkzeuge
 - Aufgaben und Arten von Testwerkzeugen
 - Automatisierungsgrad von Testwerkzeugen
 - Werkzeuge ausgewählter Testumgebungen
 - Debugger
 - Kommerzielles Beispiel: Werkzeugsuite TestBench
- ▶ 53 Werkzeuge zum Konfigurationsmanagement

- ▶ 54. Baumanagement
 - 45.1 Werkzeuge
 - 45.2 Das Baumanagementsystem von GNU
 - 45.3 Fehlermanagement
- ▶ 55. Werkzeuge für Wartung und Reengineering
 - 46.1 Aufgaben von Wartung und Reengineering
 - 46.2 Vorgehen des Reengineering
 - 46.3 Werkzeuge für das Reengineering
- ▶ Teil VI: Fortgeschrittene Themen
 - 60. Hybride MDSD
 - 61. Modellmanagement

- ▶ Verstehe, wie Technikräume die Entwicklung von Software beeinflussen (Multi-technical space development)
- ▶ Verstehe, wie Software-Werkzeuge in einem Technikraum funktionieren
 - Architektur
 - Komponenten
 - Sprachen
 - Technikräume, Modellalgebren und Metasprachen
- ▶ Verstehe, wie man sie zusammensteckt (komponiert) unter Komposition der Metamodelle und Sprachen
- ▶ Verstehe Software-Entwicklungsumgebungen als zusammengesteckte Werkzeuge
- ▶ Lerne wichtige Sprachen kennen, die Werkzeuge prägen
 - mit denen man einzelne Werkzeuge bauen kann
- ▶ Unterscheide code- und modellzentrierte Werkzeuge
- ▶ Verstehe Werkzeuge für den Software-Entwicklungsprozess

- [1 B93] Balzert, H. (Hrg.) u.a.: CASE - Systeme und Werkzeuge; BI-Wissenschaftsverlag Mannheim, 5. vollst. überarb. Auflage 1993
- [2 BAL, BOO] Balzert, H. : Lehrbuch der Software-Technik (Bd. I): Software-Entwicklung; Spektrum Akademischer Verlag (3. Auflage) Heidelberg 2009
- [3 BAL-II B98] Balzert, H. : Lehrbuch der Software-Technik (Bd. II): Software-Management. Spektrum Akademischer Verlag Heidelberg 1998
- [4 HMF] Hesse, W., Merbeth, G., Fröhlich, R.: Software-Entwicklung / Vorgehensmodelle. Projektführung, Produktverwaltung; Oldenbourg Verlag München 1992
- [5 SN92] Schönthaler, F., Nemeth, T.: Software-Entwicklungswerkzeuge: Methodische Grundlagen; B.G. Teubner Verlag Stuttgart 1992
- [6 ES89] Engels, G., Schäfer, W.: Programmmentwicklungsumgebungen - Konzepte und Realisierung; B.G. Teubner Verlag Stuttgart 1989
- [7 ZK04] Zeller, A., Krinke, J.: Open-Source-Programmierwerkzeuge (2. Auflage) dpunkt.verlag Heidelberg 2004
- [8 F91] Fisher, A.S.: CASE - Using Software Development Tools (Second Edition); John Wiley & Sons 1991
- [9 B92] Bauknecht, K. (Hrg.): Informatik-Anwendungsentwicklung - Praxiserfahrungen mit CASE; B.G. Teubner Verlag Stuttgart 1992
- [Züll] Züllighoven, Heinz. Object-Oriented Construction Handbook; dpunkt.verlag 2005
- [Raasch] Raasch. Systementwicklung mit strukturierten Methoden. Hanser. 1993

- ▶ http://en.wikipedia.org/wiki/Computer-aided_software_engineering