

Universität Karlsruhe (TH)

Forschungsuniversität · gegründet 1825

Kapitel 5

Die Implementierungsphase

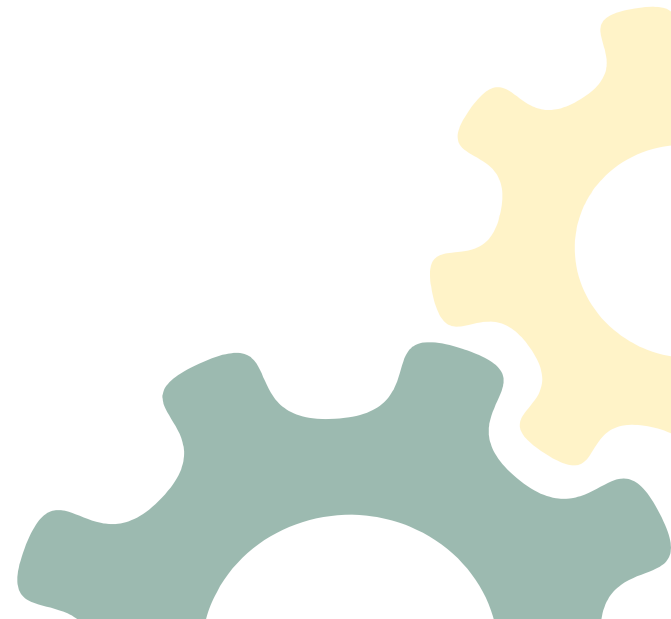
SWT I – Sommersemester 2009

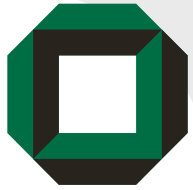
Prof. Dr. Walter F. Tichy



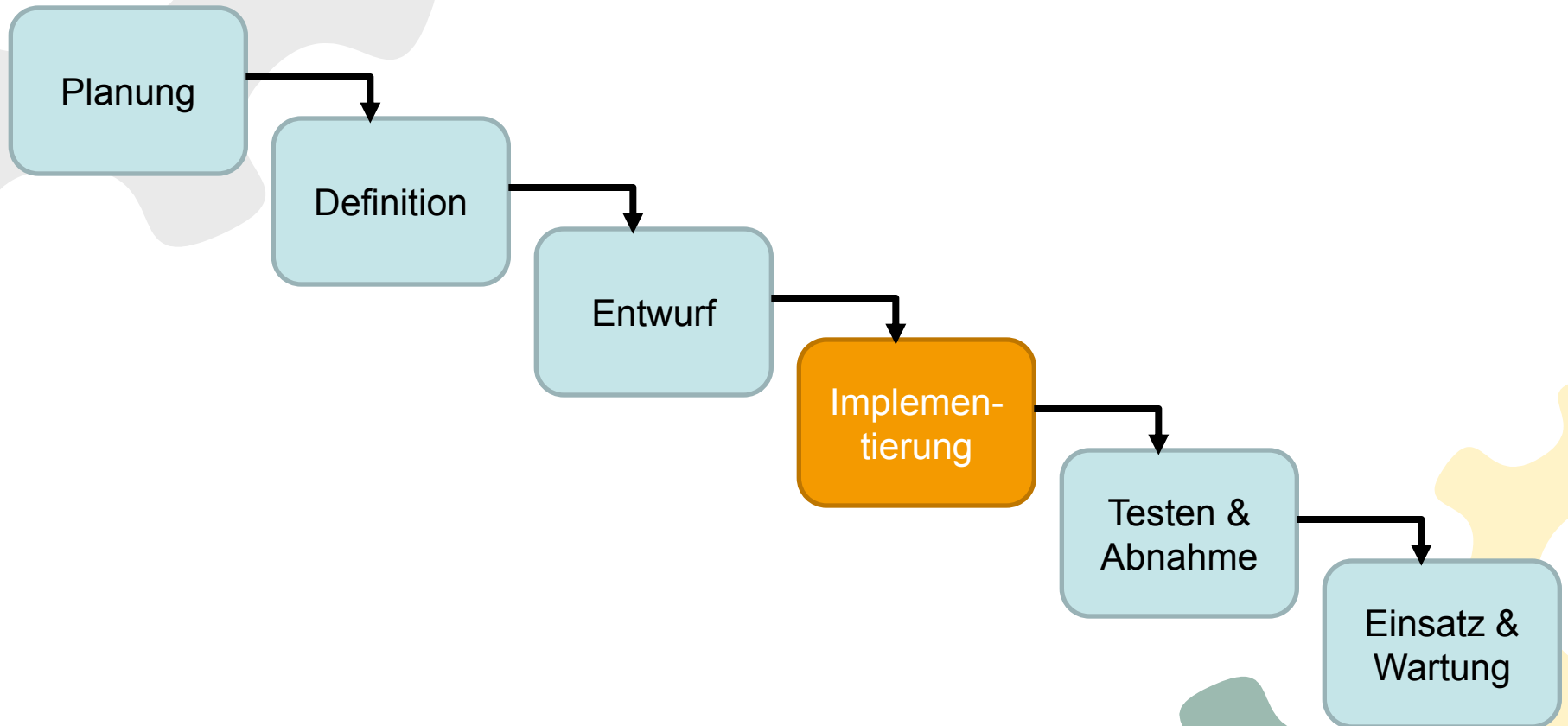
Fakultät für Informatik

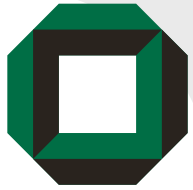
Lehrstuhl für Programmiersysteme





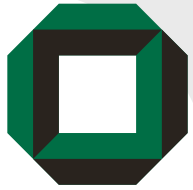
Wo sind wir gerade?





Lernziele

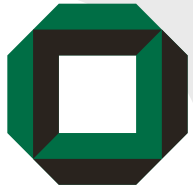
- UML in ablauffähigen Code umwandeln können.
- Programmierrichtlinien anwenden und überprüfen können.
- Selbstkontrolliertes Programmieren anwenden können.



Inhalt

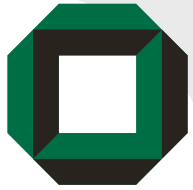
- 5. Die Implementierungsphase
 - 5.1 Einführung und Überblick
 - 5.2 Abbildung von UML auf Code
 - 5.3 Programmierrichtlinien
 - 5.4 Selbstkontrolliertes Programmieren

- siehe auch: Balzert, LE 33



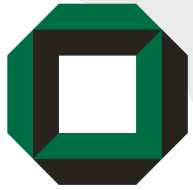
Einführung und Überblick

- Implementierungsphase:
 - **Programmierung, Dokumentierung und Testen** der Systemkomponenten aufgrund vorgegebener Spezifikationen der Systemkomponenten



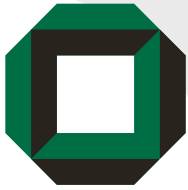
Einführung und Überblick

- Voraussetzungen
 - In der Entwurfsphase wurde eine Software-Architektur entworfen, die zu geeigneten Systemkomponenten geführt hat.
 - Abhängig von der Entwurfsmethode kann eine Systemkomponente folgendermaßen aussehen:
 - Modularer Entwurf:
 - funktionales Modul
 - Datenobjekt-Modul
 - Datentyp-Modul
 - Objektorientierter Entwurf:
 - Schnittstelle, Klasse
 - Paket (Menge von interagierenden Klassen)

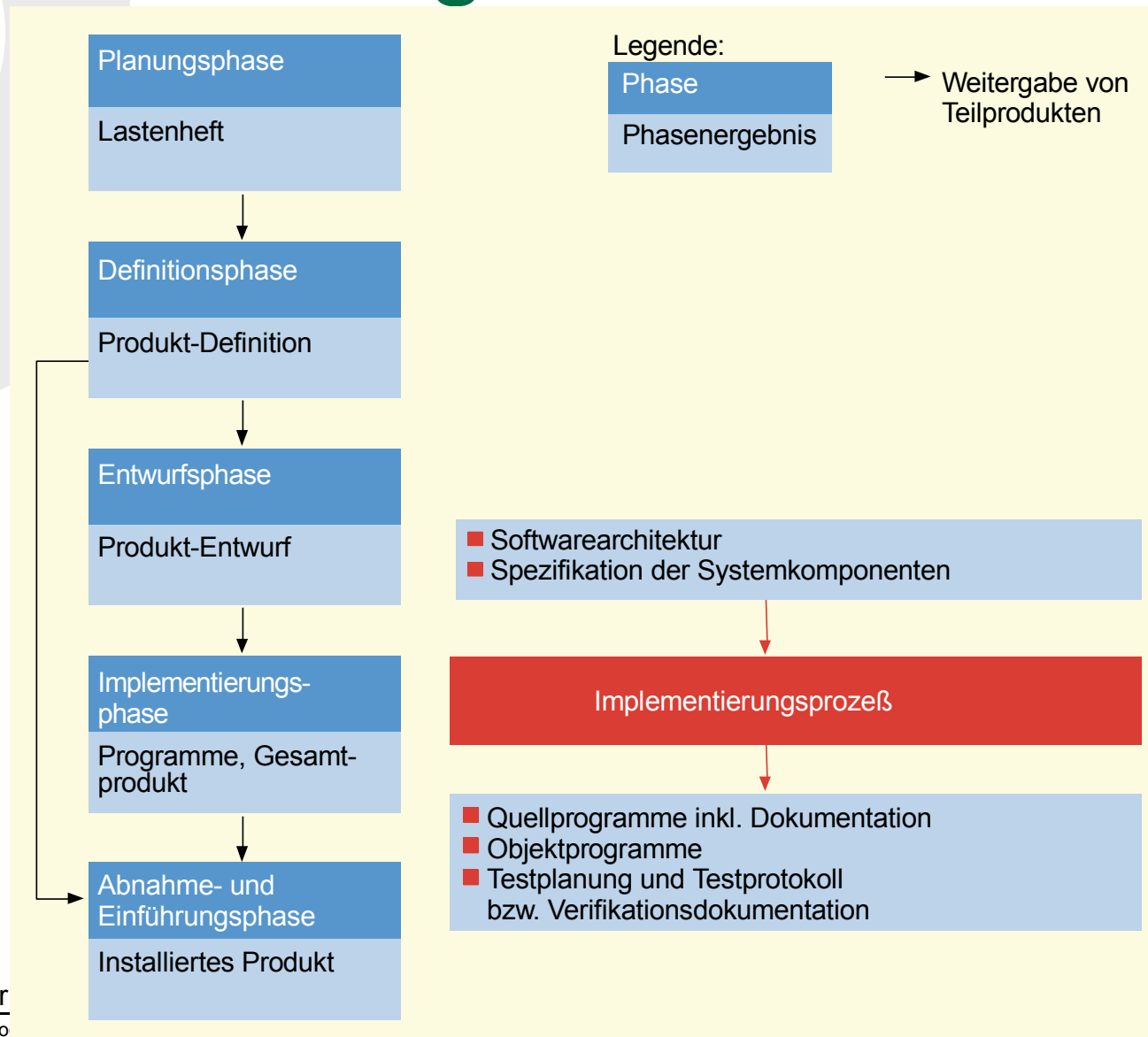


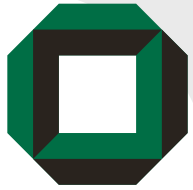
Einführung und Überblick

- Voraussetzungen
 - Für jede Systemkomponente existiert eine Spezifikation.
 - Die Softwarearchitektur ist so ausgelegt, dass die Implementierungen umfangsmäßig pro Funktion, Zugriffsoperation bzw. Methoden wenige Seiten nicht überschreiten.



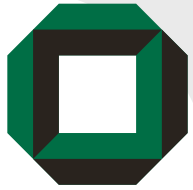
Einführung und Überblick





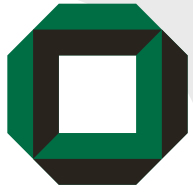
Einführung und Überblick

- Aktivitäten
 - Konzeption von Datenstrukturen und Algorithmen
 - Strukturierung des Programms durch geeignete Verfeinerungsebenen
 - Dokumentation der Problemlösung und der Implementierungsentscheidungen
 - Umsetzung der Konzepte in die Konstrukte der verwendeten Programmiersprache



Einführung und Überblick

- Angaben zur Zeit- und Speicherkomplexität
- evtl. Programmoptimierungen
- Test oder Verifikation des Programms einschl. Testplanung und Testfallerstellung
- Auch »Programmieren im Kleinen« genannt.
- Alle Teilprodukte aller Systemkomponenten müssen später integriert und einem Systemtest unterzogen werden.



Einführung und Überblick

- Teilprodukte
 - Quellprogramm einschl. integrierter Dokumentation
 - Objektprogramm
 - ausführbare Testfälle und Testprotokoll bzw. Verifikationsdokumentation