

Viewpoints in Software Architecture

Arie van Deursen¹ Christine Hofmeister² **Rainer Koschke**³
Leon Moonen¹ Claudio Riva⁴

¹CWI & Delft Univ. of Technology

²**Institut für Softwaretechnologie
Universität Stuttgart**

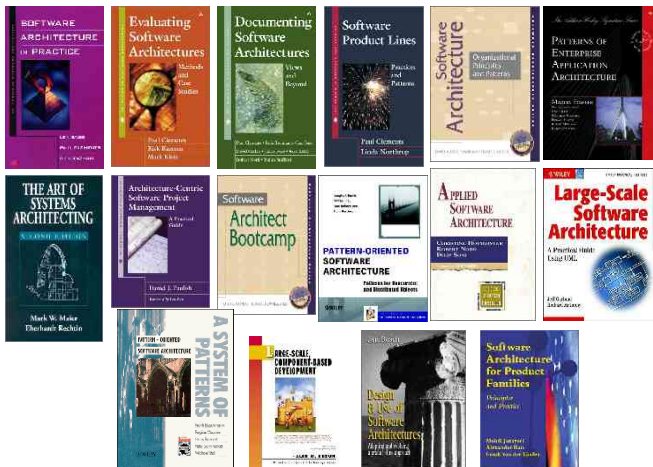
³Lehigh University

⁴Nokia Research

Workshop Software-Reengineering, Bad Honnef, 4. Mai 2004



Software-Architektur



Was ist Software-Architektur?

Definition

Software architecture is the organizational structure of a system or component [IEEE 610.12-1990].



Was ist Software-Architektur?

Definition

Software architecture is the organizational structure of a system or component [IEEE 610.12-1990].

Definition

Software architecture is the fundamental organization of a system embodied in its

- **components**,
- their **relationships** to each other and to the environment,
- and the principles guiding its design and evolution

[IEEE Recommended Practice for Architectural Description P1471-2000].



Views und Viewpoints

Definition

- A **view** is a representation of a whole system from the perspective of a set of concerns [IEEE P1471-2000].



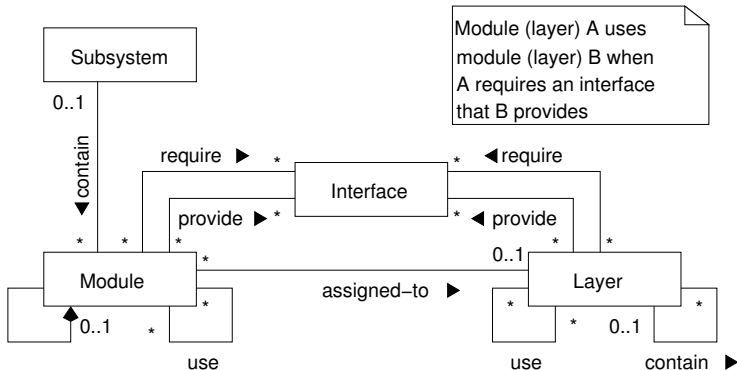
Views und Viewpoints

Definition

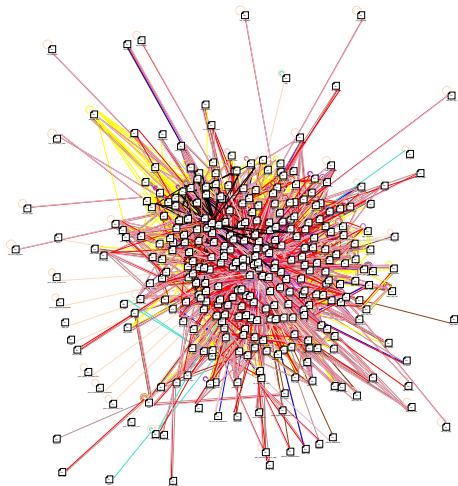
- A **view** is a representation of a whole system from the perspective of a set of concerns [IEEE P1471-2000].
- A **viewpoint** is a specification of the conventions for constructing and using a view [IEEE P1471-2000].



Beispiel: Module-Viewpoint von Hofmeister et al.



Beispiel: Module-View eines 1,5 MLOC-Systems



Viele Architekturrekonstruktionstechniken

- Tilley et al. (1996): Reverse-Engineering-Techniken folgen dem Muster:
 1. Data Gathering
 2. Knowledge Inference
 3. Information Presentation



Viele Architekturekonstruktionstechniken

- Tilley et al. (1996): Reverse-Engineering-Techniken folgen dem Muster:
 1. Data Gathering
 2. Knowledge Inference
 3. Information Presentation
- Techniken sind spezifisch für einen Viewpoint, keine übergreifende Prozessbeschreibung



Viele Architekturekonstruktionstechniken

- Tilley et al. (1996): Reverse-Engineering-Techniken folgen dem Muster:
 1. Data Gathering
 2. Knowledge Inference
 3. Information Presentation
- Techniken sind spezifisch für einen Viewpoint, keine übergreifende Prozessbeschreibung
- Keine Katalogisierung der Techniken bzgl. Viewpoints

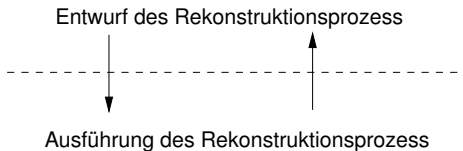


Viele Architekturekonstruktionstechniken

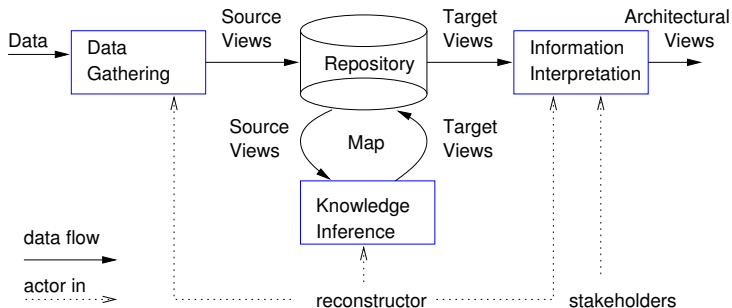
- Tilley et al. (1996): Reverse-Engineering-Techniken folgen dem Muster:
 1. Data Gathering
 2. Knowledge Inference
 3. Information Presentation
- Techniken sind spezifisch für einen Viewpoint, keine übergreifende Prozessbeschreibung
- Keine Katalogisierung der Techniken bzgl. Viewpoints
- Unser Beitrag: Symphony
 - Rekonstruktionsprozess mit Views und expliziten Viewpoints
 - Konzeptueller Rahmen für Forschung
 - Katalogisierung der Techniken/Viewpoints



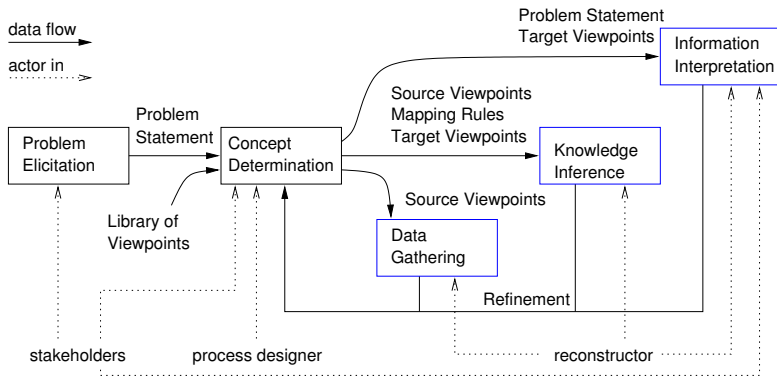
Symphony: Ebenen



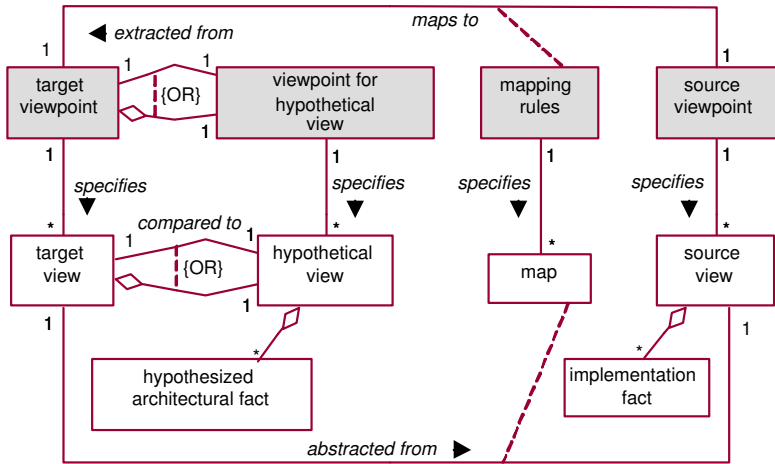
Symphony: Ausführungsebene



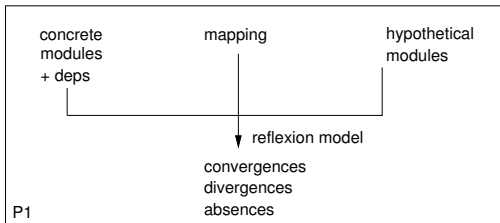
Symphony: Entwurfs- und Ausführungsebene



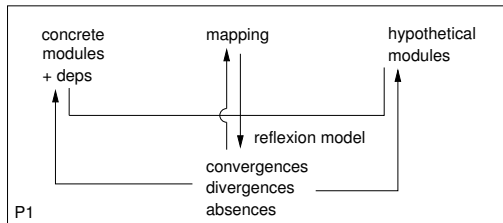
Konzeptuelles Modell von Symphony



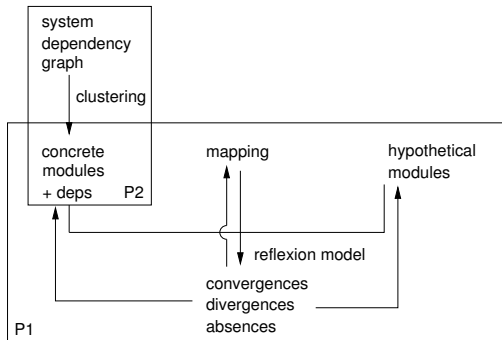
Architekturrekonstruktion als Folge von Sichten



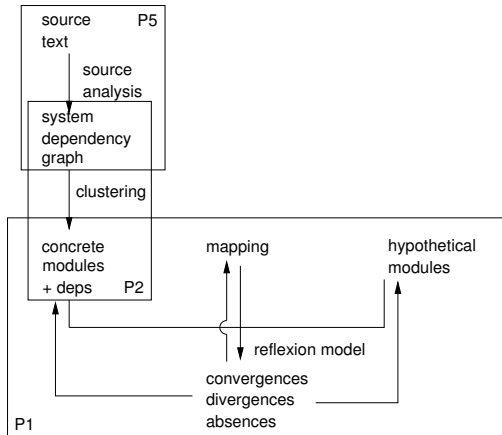
Architekturrekonstruktion als Folge von Sichten



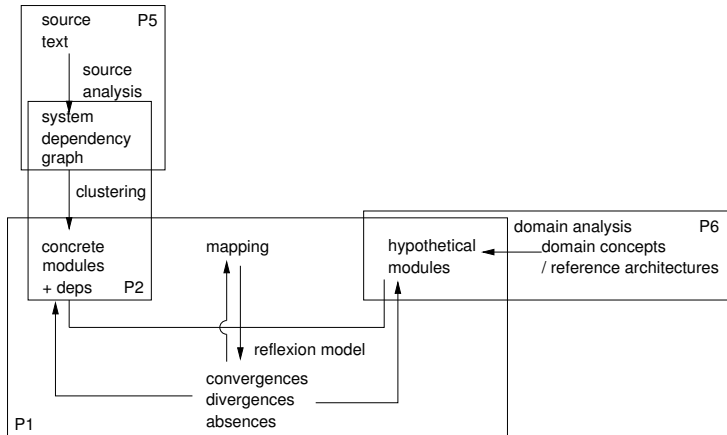
Architekturrekonstruktion als Folge von Sichten



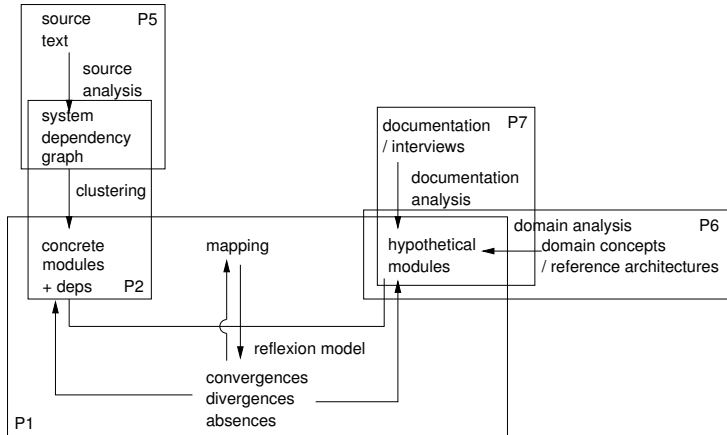
Architekturrekonstruktion als Folge von Sichten



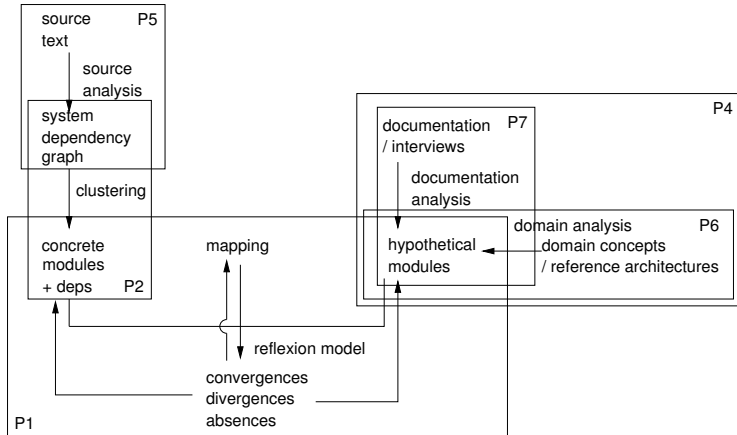
Architekturrekonstruktion als Folge von Sichten



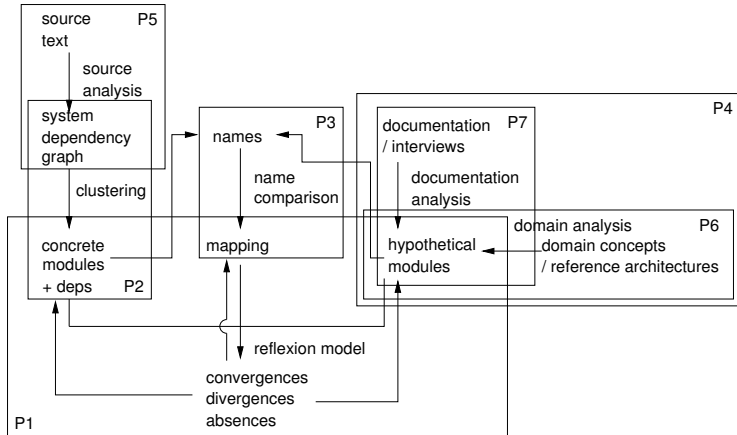
Architekturrekonstruktion als Folge von Sichten



Architekturrekonstruktion als Folge von Sichten



Architekturrekonstruktion als Folge von Sichten



Architektursichten im Forward-Engineering

	[Zachman '87]	[Perry&Wolfe '92]	[Kruchten '95]	[Hofmeister '95]	[TAFIM '95]	[Druffel '94]	[Emery '96]	[Sowa '92]	[Softtech '78]	[Moriconi '94]	[IEEE1471]
behavioral, dynamic, operational			×	×	×						×
data, data flow, information	×	×				×					×
development, maintenance	×						×				
distributed, network, physical			×		×		×	×			×
functional, activity	×	×						×	×		
logical, conceptual			×	×	×					×	
decomposition, module, static			×	×							
code				×							



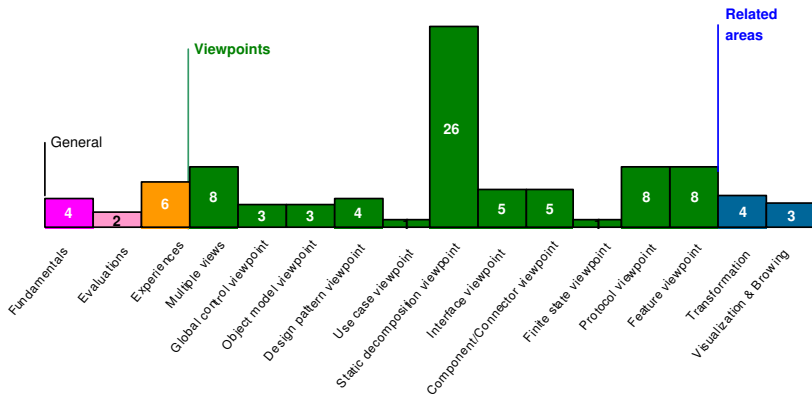
Architektursichten im Reverse-Engineering

Literaturstudie:

- IEEE
 - WCRE 1996 - 2000
 - ICSM 1996 - 2000
 - IWPC 1998 - 2000
 - CSMR 1997 - 2000
 - Trans. SE 1995 - 2001
- ACM
 - ICSE 1987 - 2000
 - PASTE 1998 - 1999
 - TOSEM 1992 - 2001
 - SIGSOFT 1990 - 2000



Architektursichten im Reverse-Engineering



Symphonys Konsequenzen für die Forschung/Praxis

- Gliederung der Techniken/Methoden in Aktivitäten (Data Gathering, Knowledge Inference, Information Interpretation)
 - Views bilden den Informationsfluss zwischen Aktivitäten



Symphonys Konsequenzen für die Forschung/Praxis

- Gliederung der Techniken/Methoden in Aktivitäten (Data Gathering, Knowledge Inference, Information Interpretation)
 - Views bilden den Informationsfluss zwischen Aktivitäten
- Beschreibung der Techniken/Methoden durch Viewpoints
 - Viewpoints beschreiben die Schnittstelle zwischen Aktivitäten (“Prozessinteroperabilität”)
 - Katalogisierung der Techniken/Methoden in die adressierten Viewpoints

