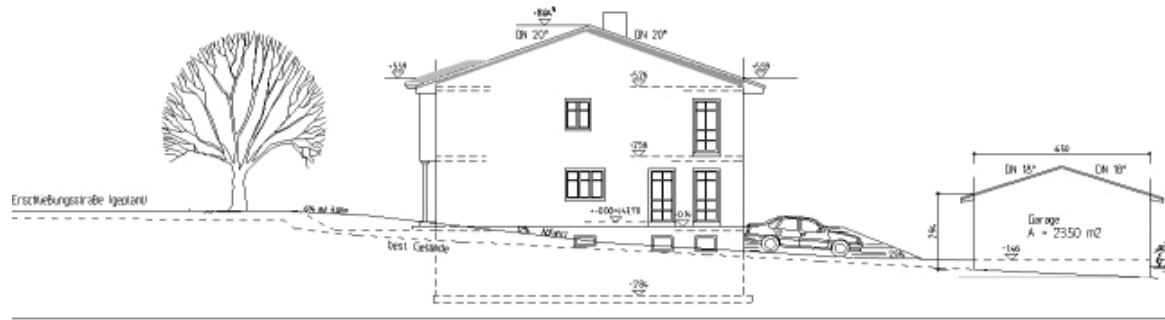


Architektursichten nach ARC42

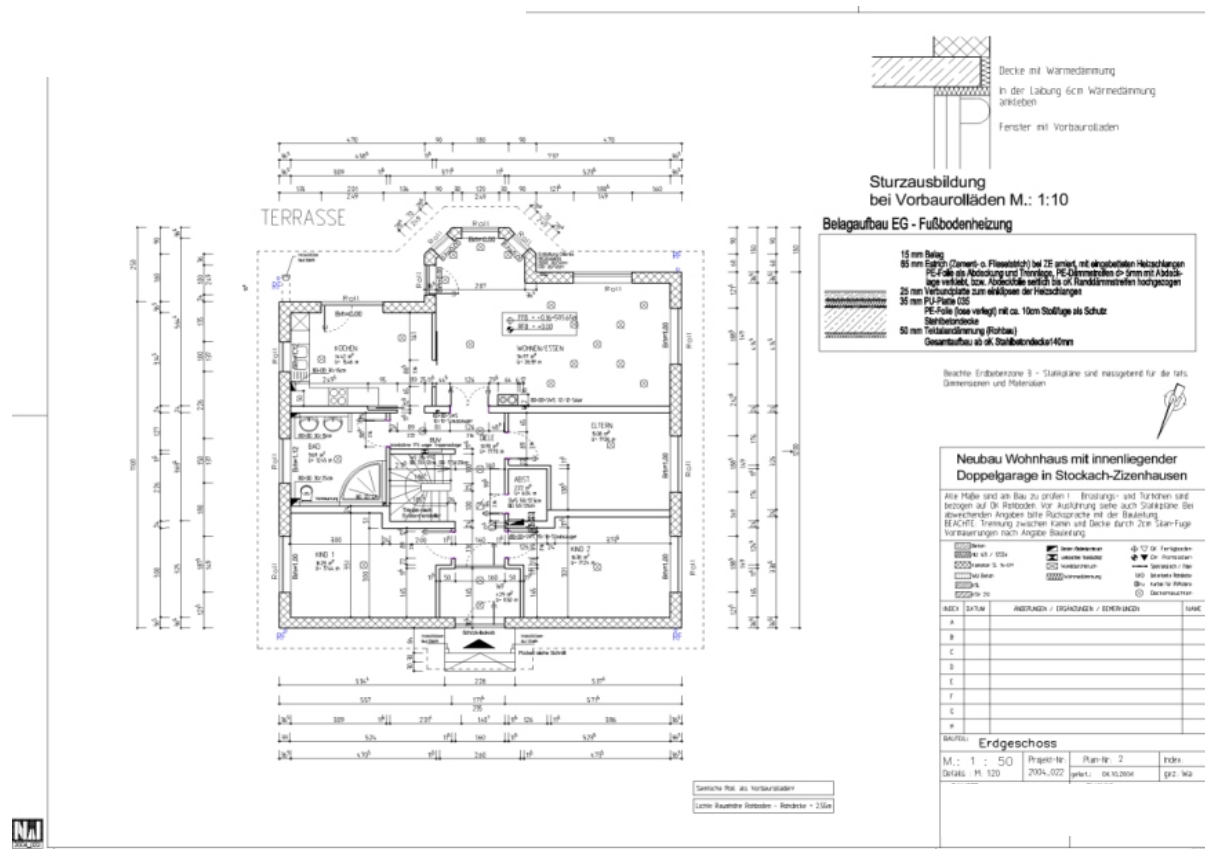
- Perspektiven / Sichten, engl. *Views*
- Drei Sichten nach ARC42
 - (Abstraktes) Beispiel

Motivation für verschiedene Sichten

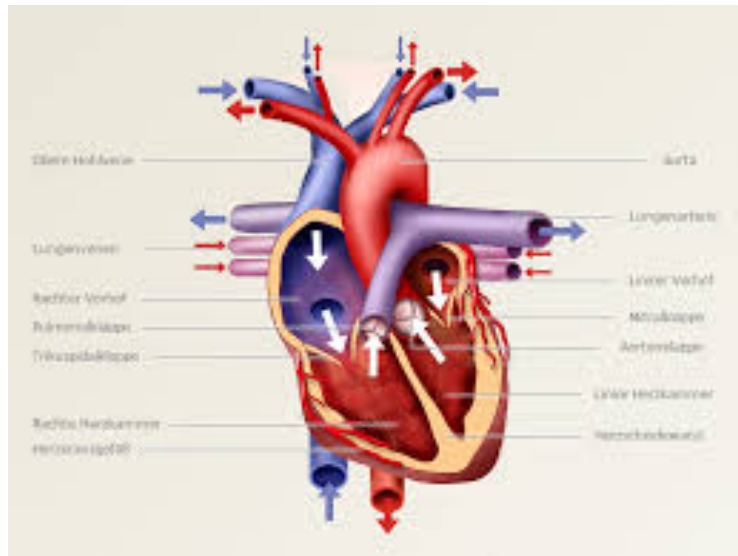


<http://www.planungswerkstatt-bau.de>

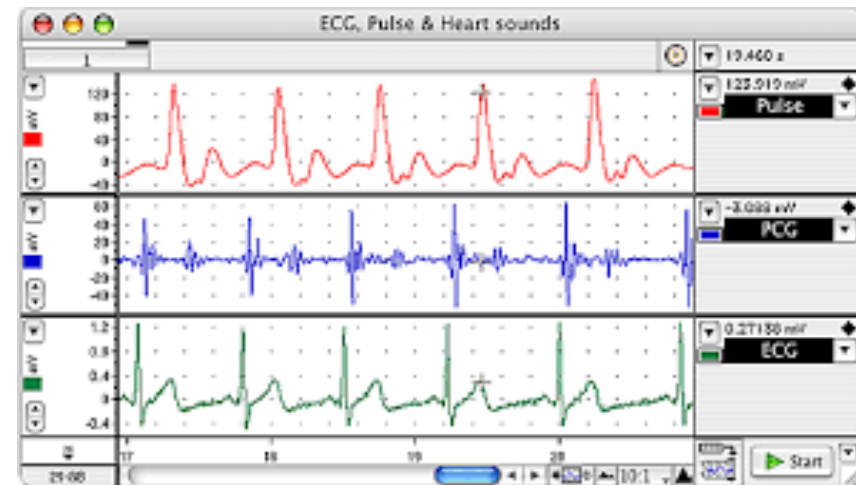
ANSICHT AUS SÜDEN



One human body has many structures, and those structures have many views.
So it is with software...

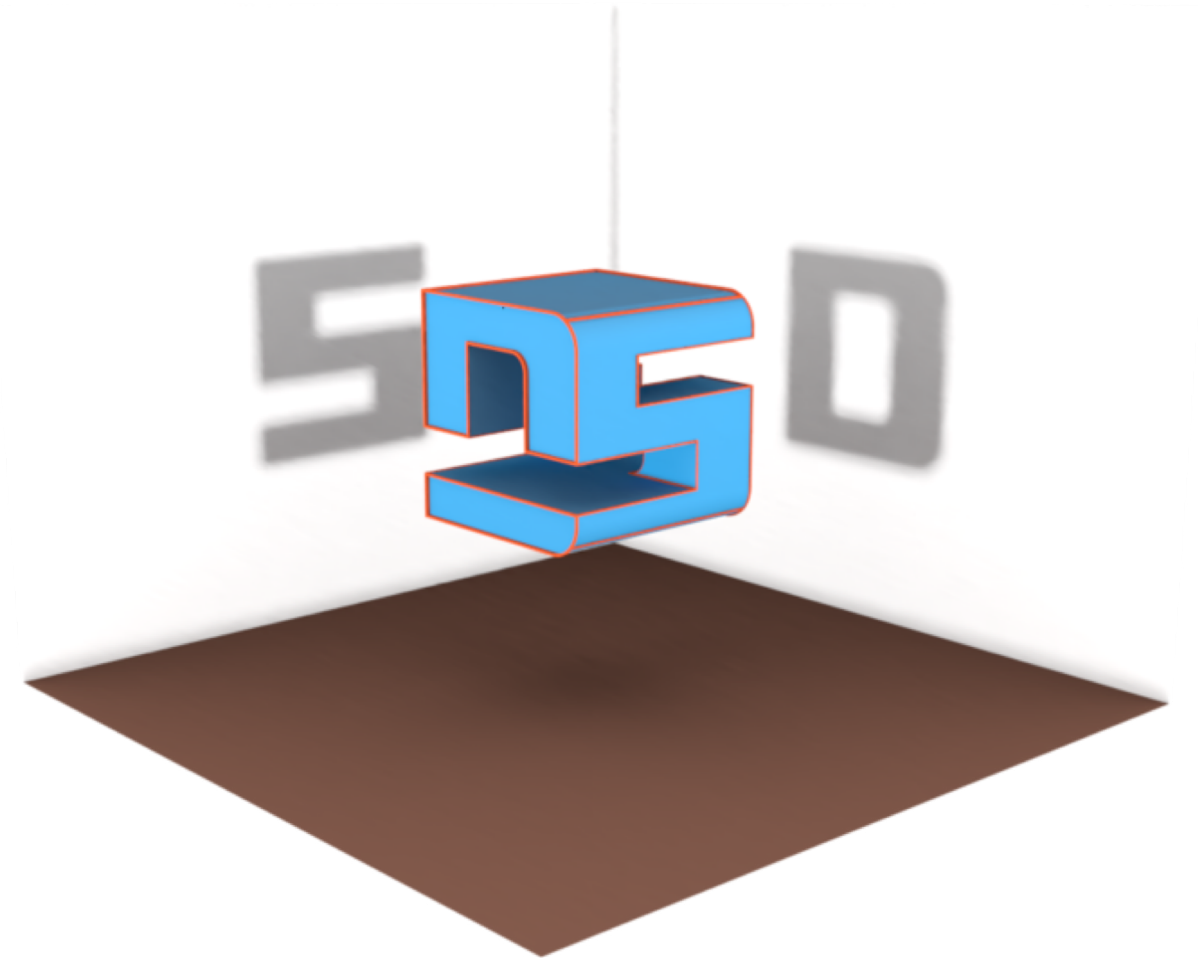


Statische Struktur

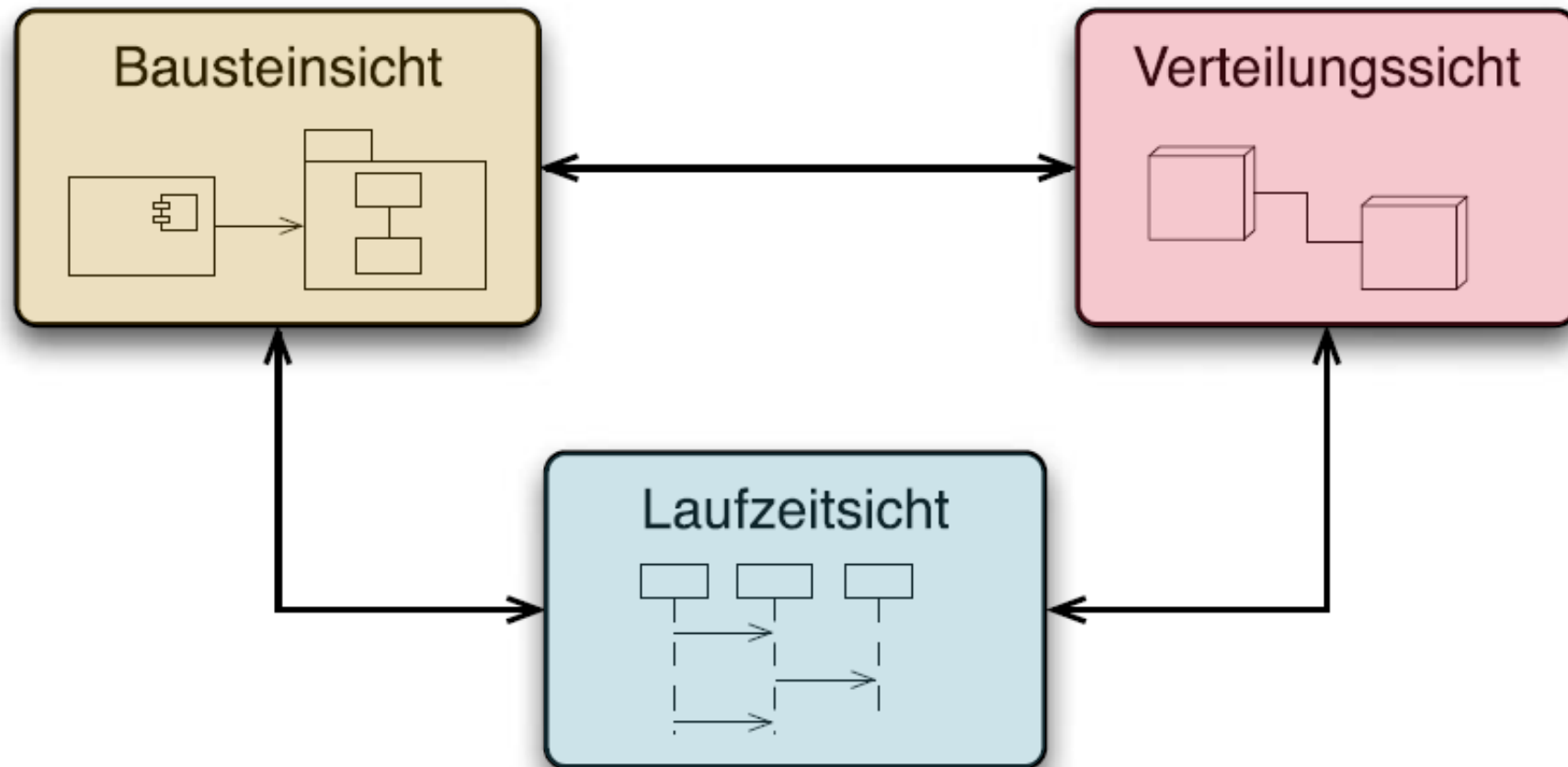


Dynamisches Verhalten

Statische und dynamische Sicht: *dieselbe Architektur* aus mehreren Perspektiven



Sichten nach ARC42 (Starke, Hruschka) 1/2



Gernot Starke, Peter Hruschka: Software-Architektur kompakt, 2009

Vorteile verschiedener Sichten

- Separation of Concerns
 - Analyse / Diskussion einzelner (getrennter) Aspekte möglich
- Reduktion von Komplexität
 - Konzentrieren auf bestimmte Belange
- Kommunikation mit unterschiedlichen Stakeholdern
 - Vor allem Entwickler... konzentrieren sich gerne auf implementierungsrelevante Teile
 - Andere Stakeholder verstehen andere Dinge, benötigen andere Informationen

Sichten nach ARC42 (Starke, Hruschka) 2/2

1. Kontextabgrenzung

- mit welchen anderen Systemen / welchen Stakeholdern interagiert das System
- Ergebnis der Anforderungsanalyse

2. Bausteinsicht

- wie ist das System aus Softwarebausteinen aufgebaut (*statische* Struktur)
- deren Beziehungen und Schnittstellen untereinander

3. Laufzeitsicht

- wie interagieren die Bausteine des Systems zur Laufzeit miteinander bzw. mit den Nachbarsystemen (*dynamisches* Verhalten)
- wesentliche, beispielhafte Abläufe

4. Verteilungssicht

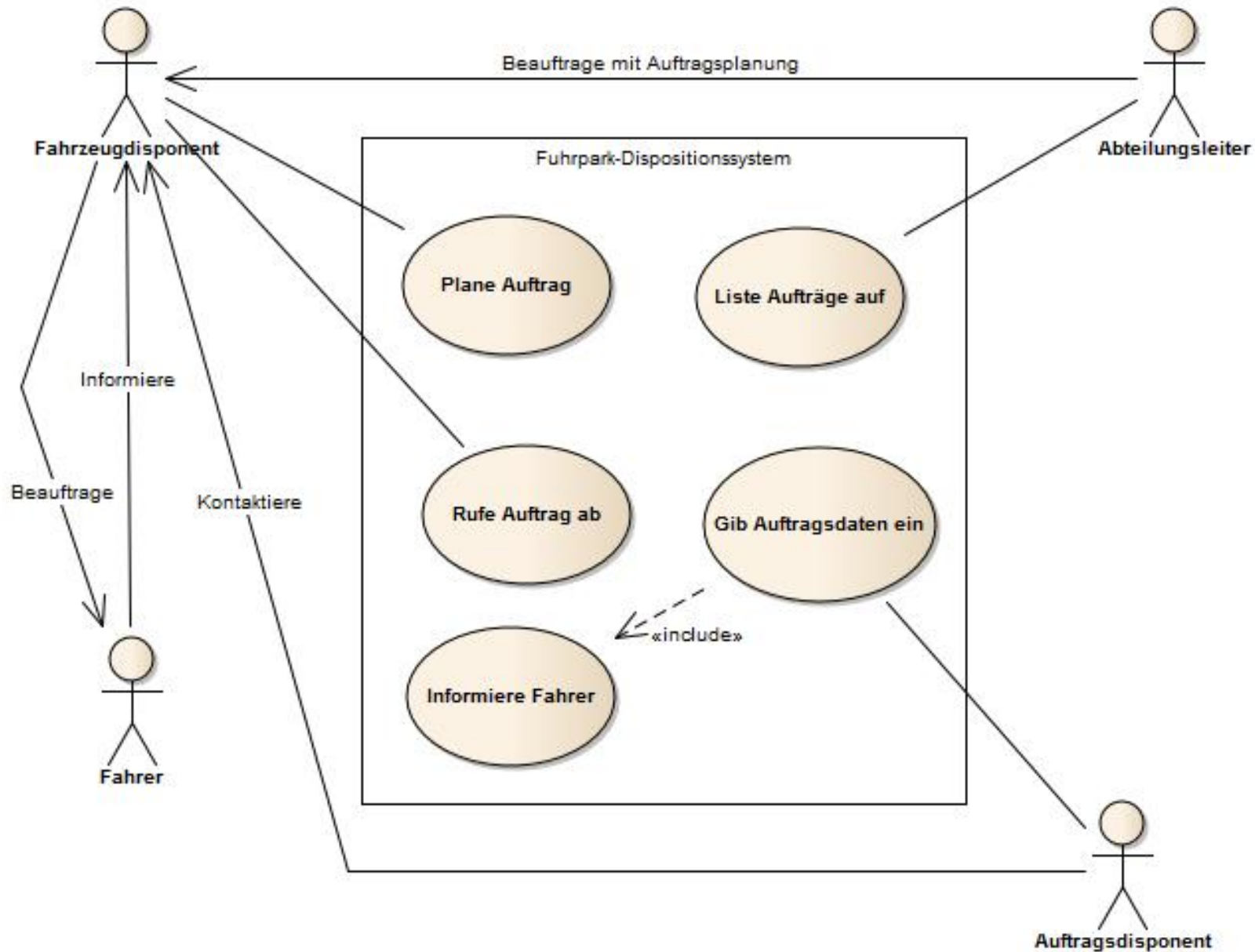
- welche Teile des Systems laufen auf welchen Rechnern
- an welchen geographischen Standorten
- in welcher Umgebung

Architektur-Dokumentation: arc42

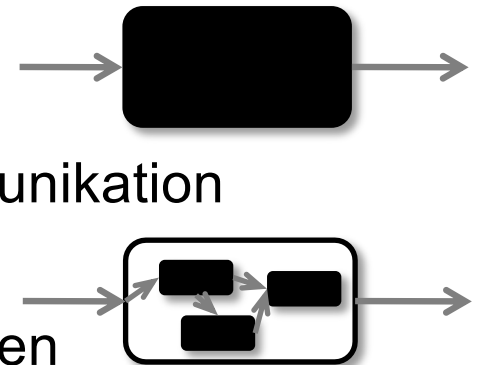
1 Einführung und Ziele 1.1 Übersicht fachlicher Anforderungen 1.2 Wesentliche Architekturziele 1.3 Projektbeteiligte (Stakeholder)	Kurze Zusammenfassung der wesentlichen Anforderungen an das System. Wichtig ist die Darstellung der nichtfunktionalen Anforderungen (Qualitätsmerkmale), die i.d.R. die wesentlichen Architekturziele bilden.
2 Randbedingungen 2.1 Technische Randbedingungen 2.2 Organisatorische Randbedingungen 2.3 Konventionen	Faktoren, die den Systemarchitekten beim Entwurf einschränken.
3 Kontextabgrenzung 3.1 Fachlicher Kontext 3.2 Technischer oder Verteilungskontext	Die Top-Level Darstellung des Systems in seinem fachlichen oder technischen Umfeld.
4 Bausteinsicht	Die statische Struktur, die Zerlegung des Systems in Implementierungsbestandteile, hier Bausteine genannt. Beschrieben in verschiedenen Verfeinerungsebenen.
5 Laufzeitsicht	Zeigt die Zusammenarbeit von Bausteinen zur Laufzeit, illustriert die Dynamik des Systems.
6 Verteilungssicht	Beschreibt die möglichen Ausführungsumgebungen des Systems, seine technische Infrastruktur.
7 Typische Muster, Strukturen und Abläufe	Wiederholt benutzte oder auftretende Strukturen, die sich in die Top-down-Zerlegungen von Kapitel 4 und 6 nicht systematisch einfügen lassen.
8 Übergreifende Architektur Aspekte und technische Konzepte	Lösungsansätze für übergreifende Themen, wie Persistenz, Transaktionen oder GUI.
9 Entwurfsentscheidungen	Zentrale architekturelevante Entscheidungen mit Begründung

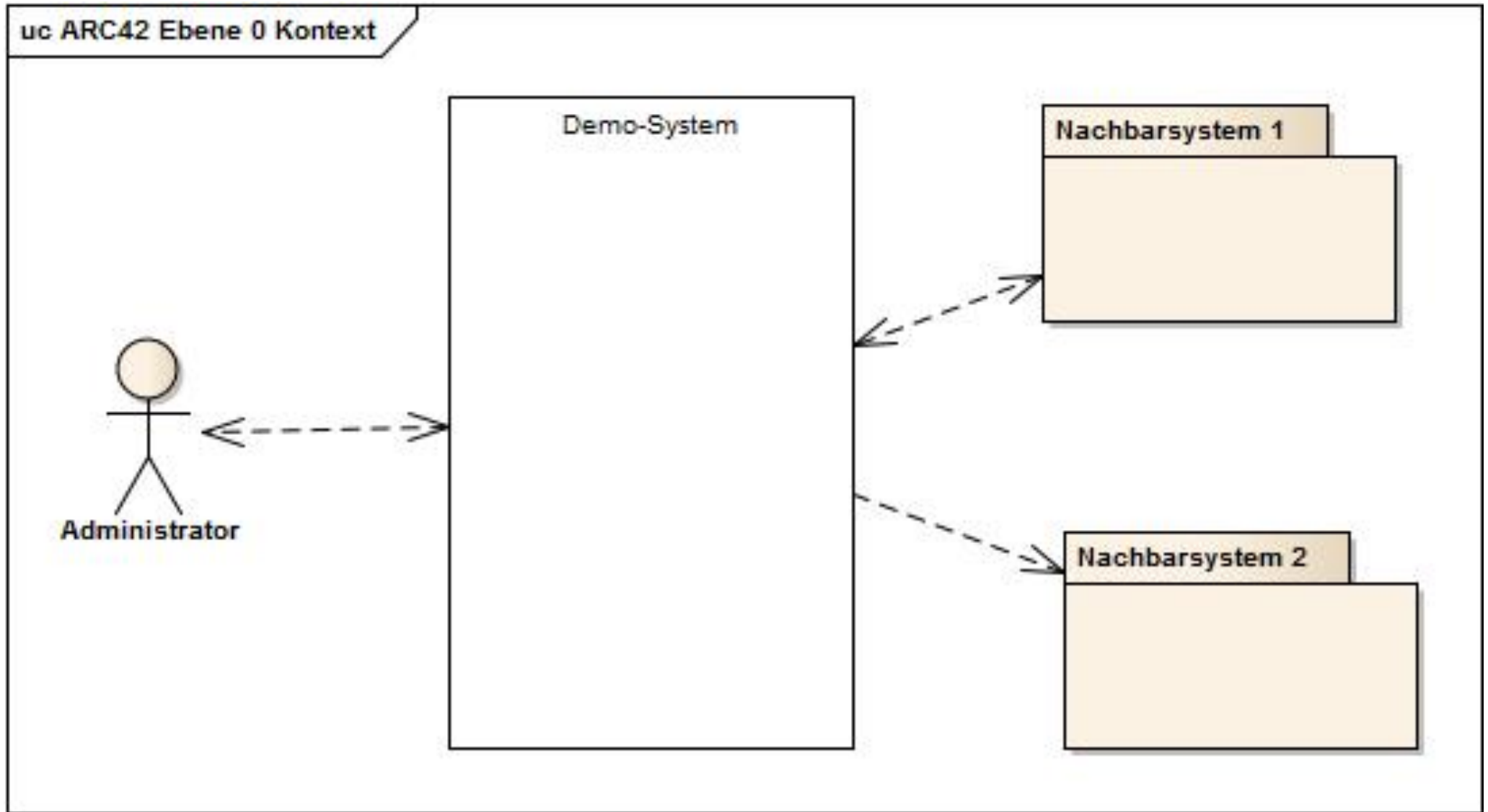
Quelle: arc42 template, <http://www.arc42.de/downloads>

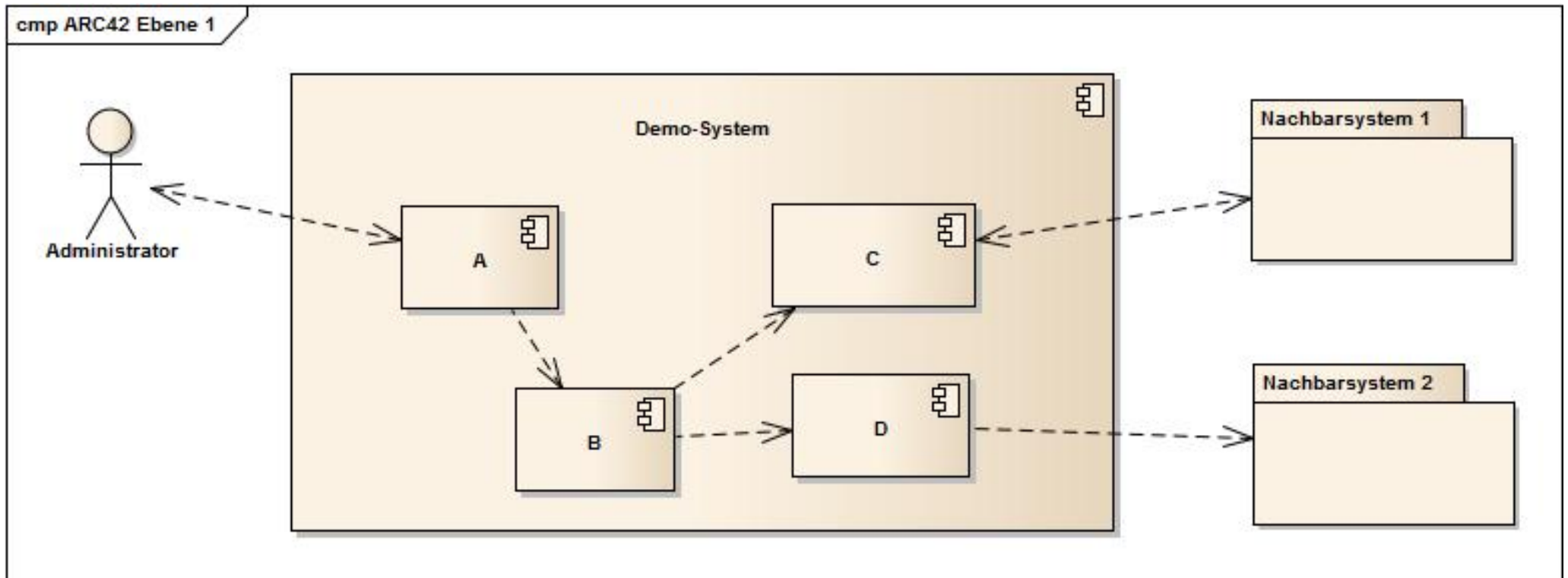
Kontextabgrenzung: Use Case-Diagramm (empfohlen)



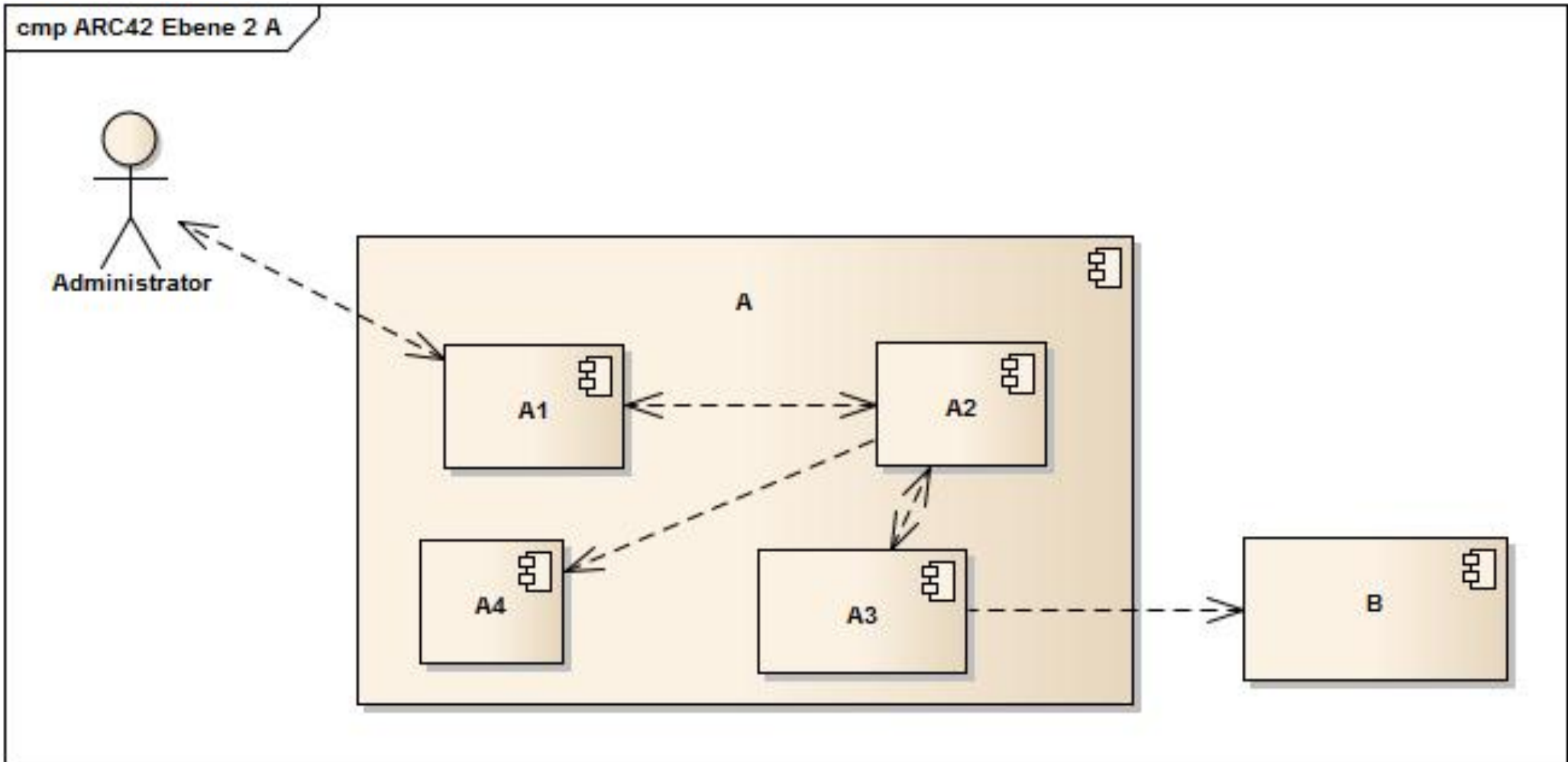
- UML-Diagramme
 - Komponenten-, Paket- oder Klassendiagramm
- Hierarchische Verfeinerung des Systems
 - Tiefe abhängig von Erfordernissen
 - Ebene 0 = Wurzel: Kontextdiagramm
 - ▶ Kommunikation mit externen Komponenten
 - ▶ Keine Darstellung von System-interner Kommunikation
 - Ebene 1: system-interne Komponenten
 - ▶ Zerlegung des Gesamtsystems in Komponenten
 - Ebene 2: Zerlegung der Komponenten aus Ebene 1
 - Ebene 3: Zerlegung der Komponenten aus Ebene 2
 - Beim Übergang in einer Ebene zur anderen müssen die Schnittstellen gleich bleiben!

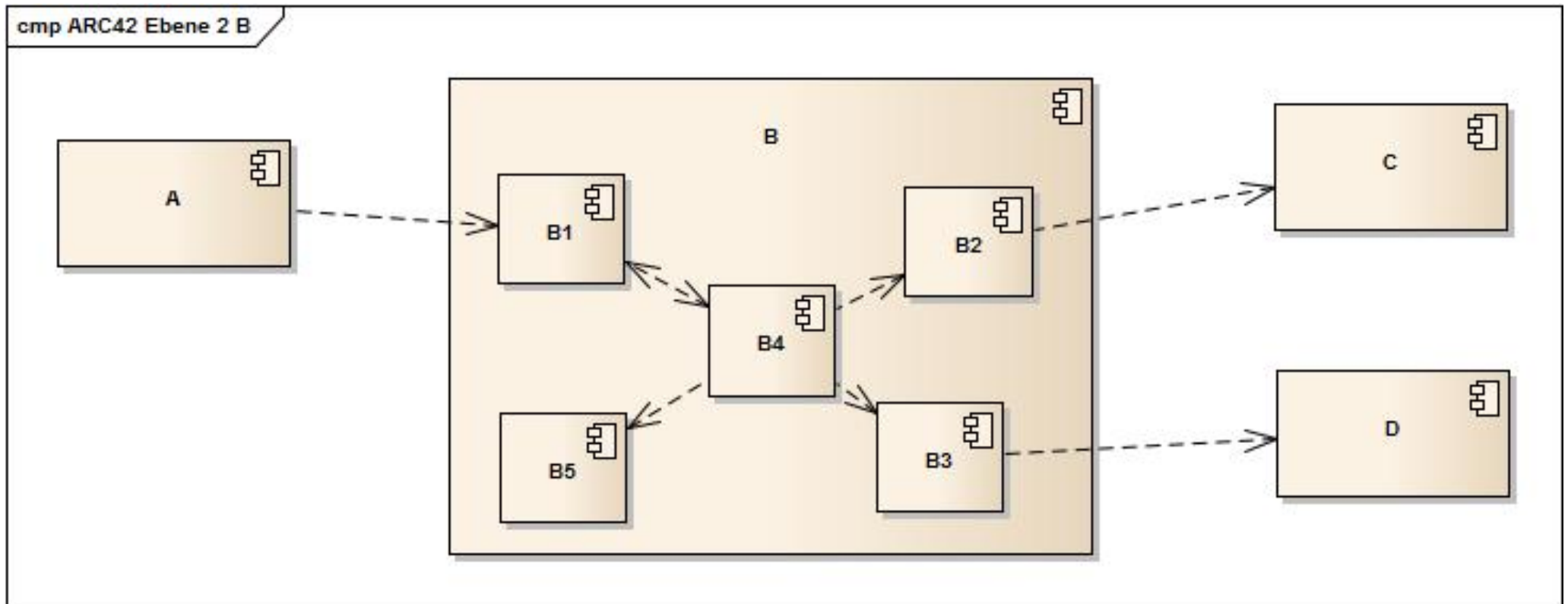




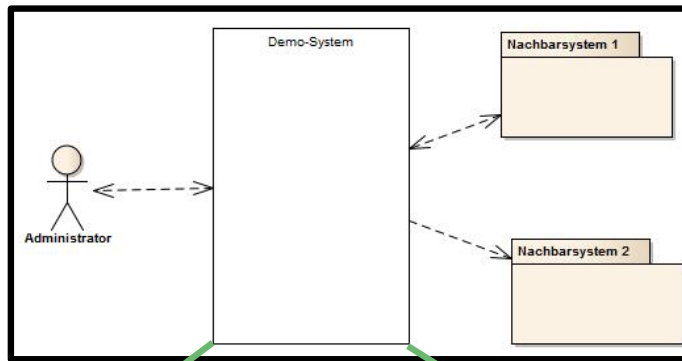


cmp ARC42 Ebene 2 A



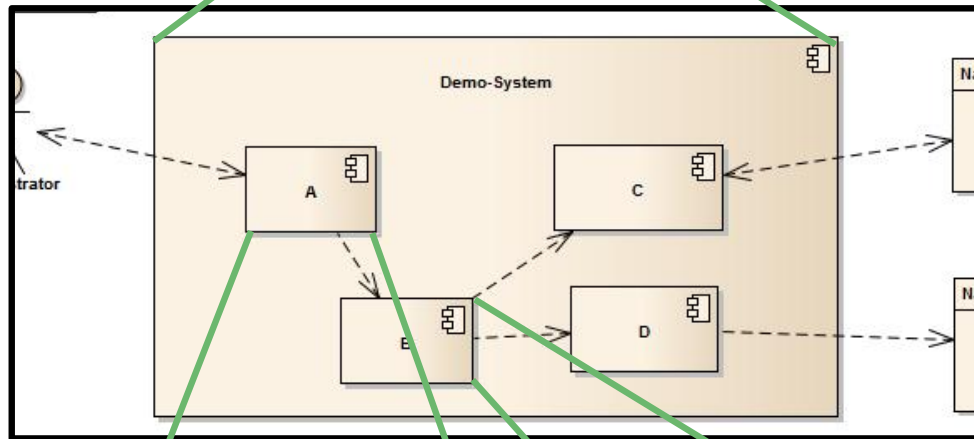


Die Ebenen im Überblick



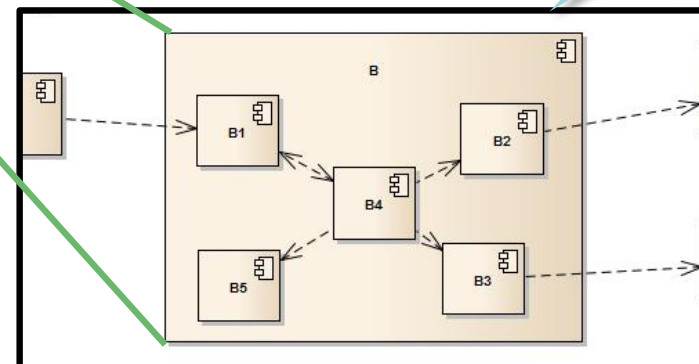
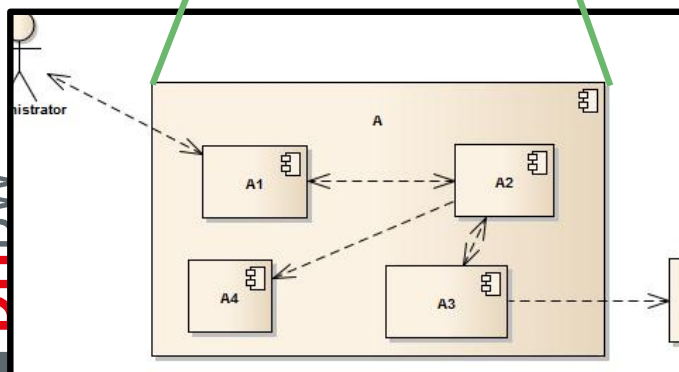
(aus der Anforderungsspezifikation)

System als Blackbox

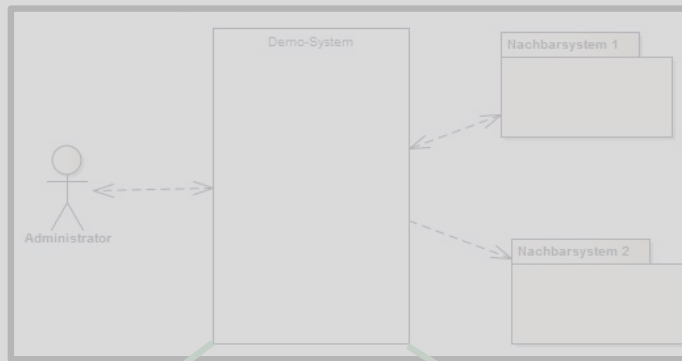


1. Verfeinerungsebene

2. Verfeinerungsebene



Die Ebenen
im Überblick



(aus der
Anforderungs-
spezifikation)

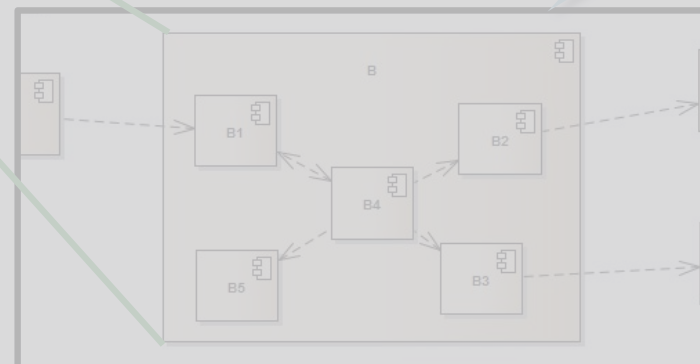
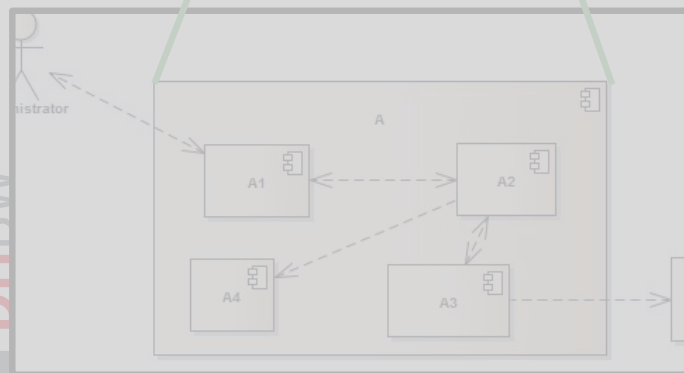
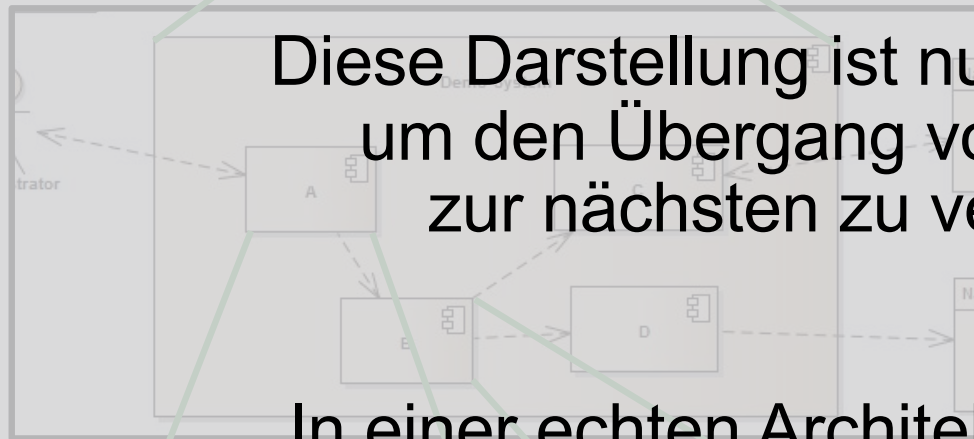
System als
Block

Diese Darstellung ist nur eine Illustration,
um den Übergang von einer Ebene
zur nächsten zu verdeutlichen.

1. Verfeinerungs-
ebene

In einer echten Architekturbeschreibung
sind das vier verschiedene Diagramme!

2. Verfeinerungs-
ebene



- UML-Diagramm
 - Deployment-Diagramm
 - Verfeinerung
 - Ebene 0: technisches Kontextdiagramm
 - ▶ Kommunikation mit externen Systemen
 - ▶ Deployment-Diagramm mit dem System (= ein Knoten)
 - Ebene 1: system-intern
 - ▶ Hardware-Komponenten mit zugehörigen Software-Komponenten
- Beispiele:
- variable Anzahl von Servern
 - DB und Server zusammen / verteilt

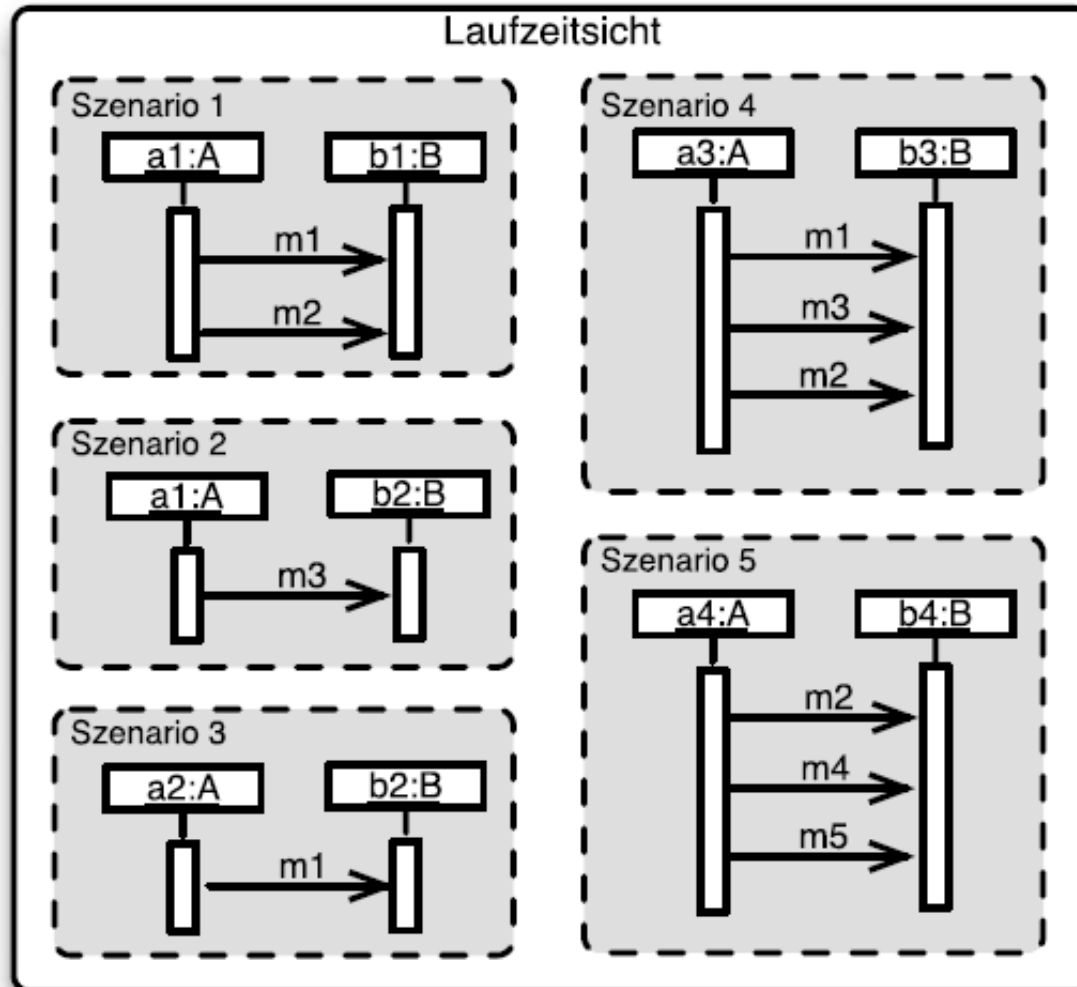


geeignet fürs
Pflichtenheft

-> Es werden überall (nur) die Komponenten der Bausteinsicht deployt!

- UML-Diagramme
 - Sequenzdiagramm
 - Manchmal: Kommunikationsdiagramm, Aktivitätsdiagramm
- Verfeinerung
 - Ebene 0:
 - ▶ Kommunikation mit externen Systemen aus dem Kontextdiagramm
 - Ebene 1: system-interne Zusammenarbeit
 - ▶ 1. Darstellung
 - Kommunikation zwischen Hardware-Komponenten (notwendig: Verteilungssicht)
 - ▶ 2. Darstellung
 - Kommunikation zwischen Software-Komponenten
 - Ebene 2 ...
 - ...

Laufzeitsicht 2/2



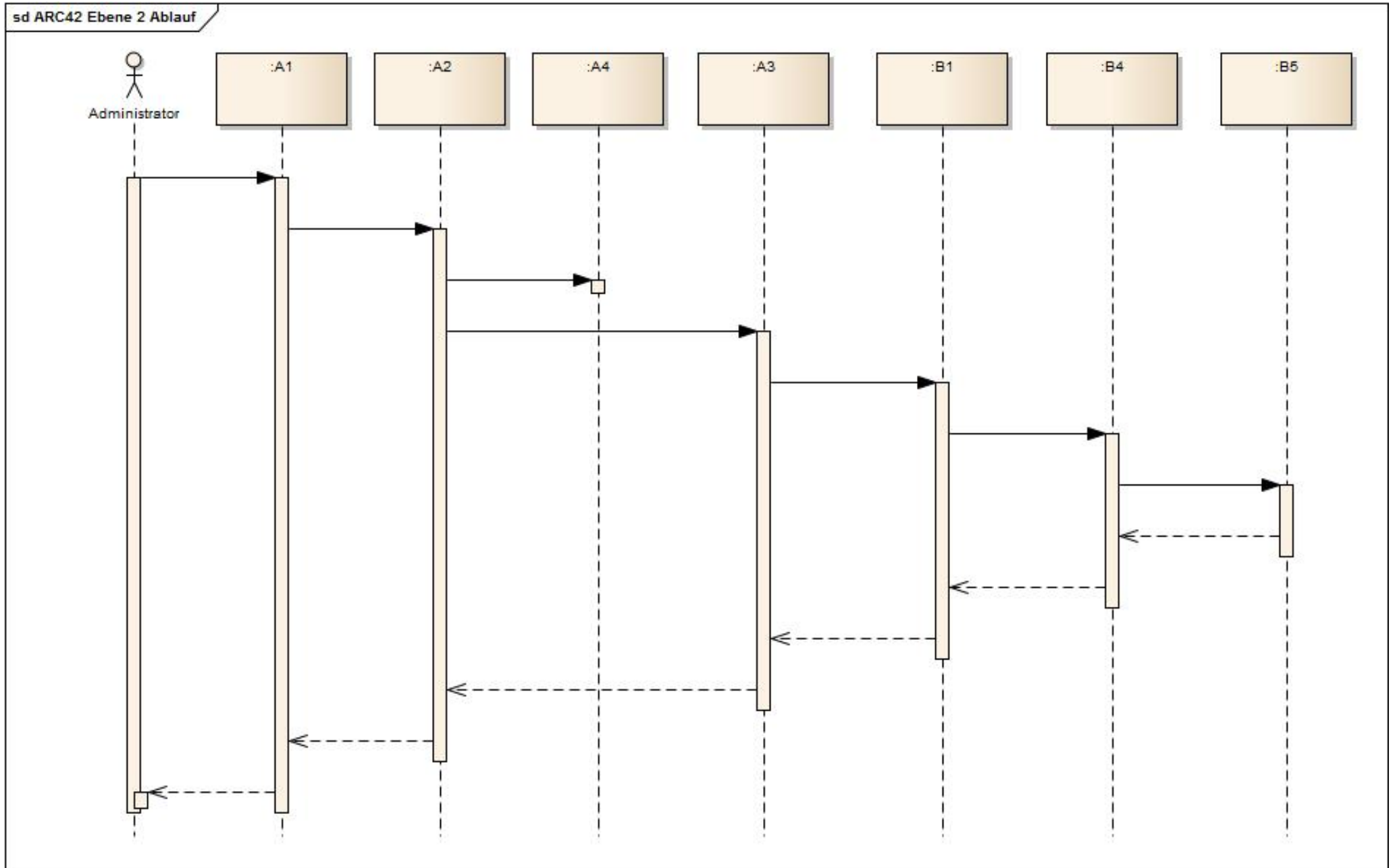
Darstellung ausgewählter Szenarien

- *typische* Abläufe
- *komplexe* Abläufe
- *unklare* Abläufe

Nicht dargestellt werden

- ähnliche Abläufe
- offensichtliche Abläufe

Laufzeitsicht am Beispiel von vorhin



Laufzeitsicht am Beispiel von vorhin

