Software-Engineering

Vorlesung 7 vom 29.11.2004

Sebastian Iwanowski
FH Wedel

Software-Engineering

Vorlesungsthemen:

1. Überblick über das Thema und die Vorlesung
2. Grundlegende Prinzipien
3. Softwareplanung
4. Systemanalyse
5. Softwareentwurf
6. CASE-Tools
7. Aufwandsabschätzung
8. Qualitätsmanagement
9. Projektmanagement

Bestandteile eines UML-Tools

Die wichtigsten Funktionalitäten von Rational Rose und ähnlichen Werkzeugen:

- objektorientierte Systembeschreibung: Klassendiagramme mit Beziehungen und Hierarchien
- Beschreibung von Anwendungsszenarien (Use Cases)
- ➔ Beschreibung von Prozessabfolgen (Sequenzdiagramme)
- Beschreibung von Zustandsabfolgen (Zustandsübergangsdiagramme)
- Aktivitätsdiagramme: spezielle Zustands-Aktivitäts-Beschreibungen

UML: Sequenzdiagramme

zur Beschreibung von Transaktionen / Interaktionen

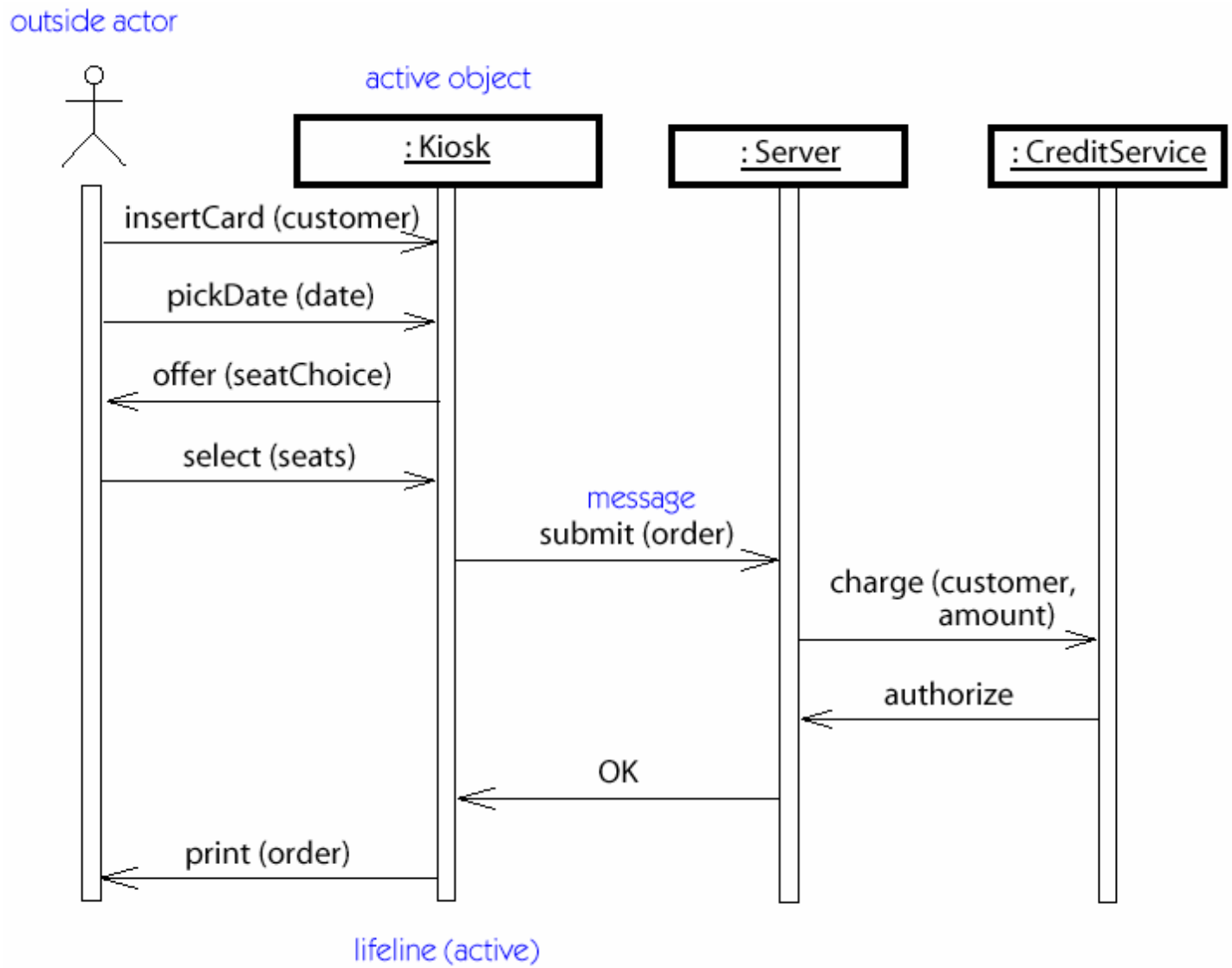
Transaktion:

- Menge von Nachrichten, die zwischen einer Gruppe von Objekten ausgetauscht werden, um eine bestimmte Operation oder ein bestimmtes Ergebnis zu veranlassen.
- In vielen Anwendungsfällen müssen Transaktionen unteilbar sein und weitere Anforderungen erfüllen (z.B. im Anwendungsfall Datenbanken)

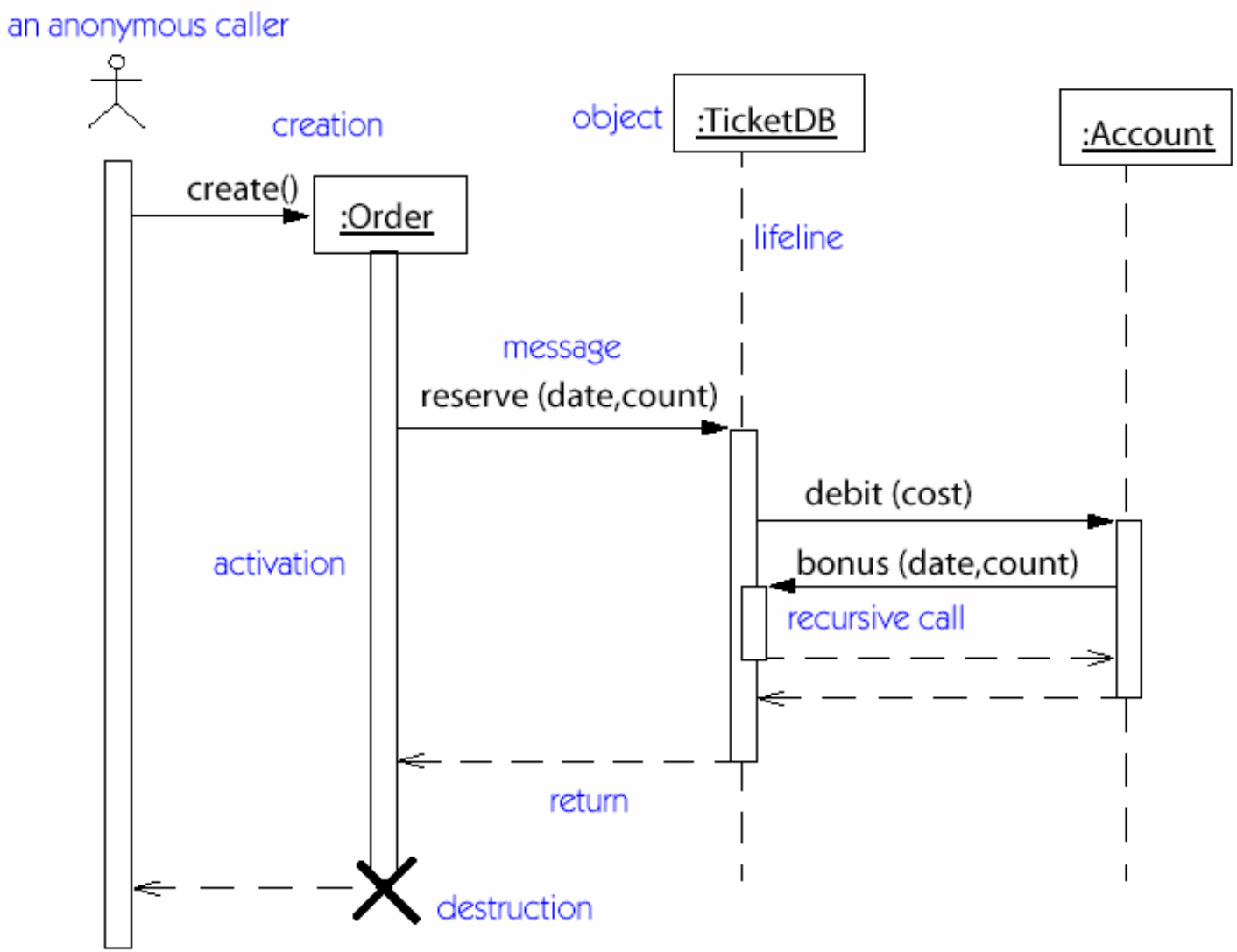
Interaktion:

- Der Austausch einer einzelnen Nachricht

Sequenzdiagramme: 1. Beispiel



Sequenzdiagramme: 2. Beispiel

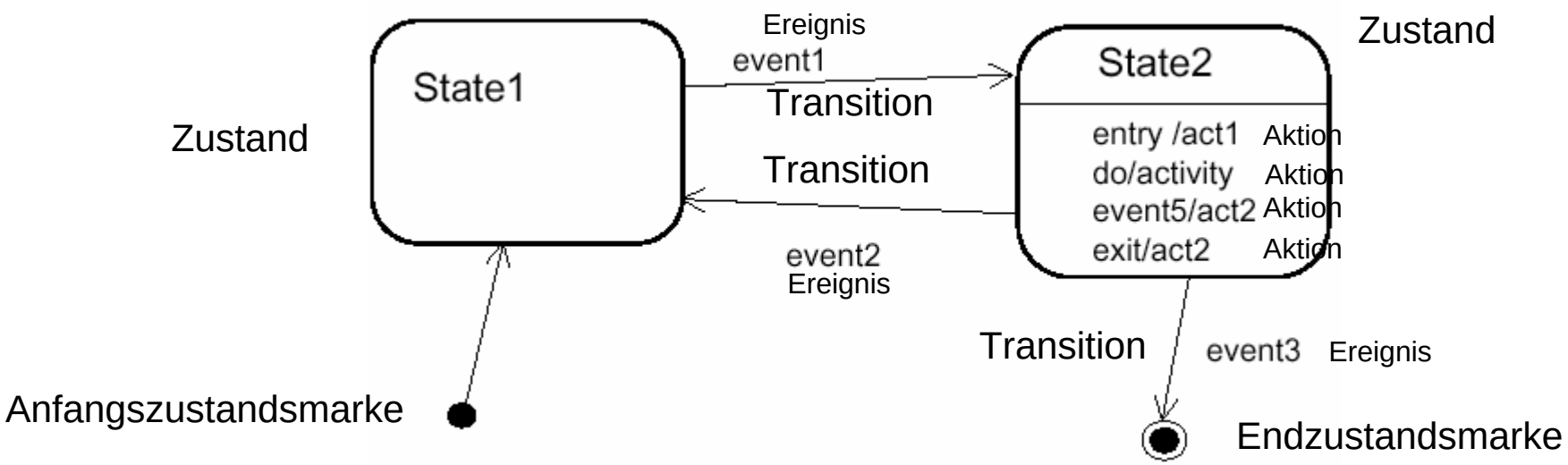


UML: Zustandsübergangsdiagramme (Statecharts)

zur Beschreibung von dynamischen Objekten des Systems

Zustandsübergangsdiagramme beschreiben:

- die Lebenshistorie von Objekten einer Klasse
- die Ereignisse, die einen Übergang (Transition) von einem Zustand zu einem anderen veranlassen
- die Aktionen, die sich aus diesem Zustandswechsel ergeben



UML: Zustandsübergangsdiagramme (Statecharts)

Klassifizierung der Aktionen:

Generell wird eine Aktion bezeichnet durch: **`Ereignis / Aktion`**

entry-Aktionen

- werden immer ausgeführt beim Betreten des Zustandes

exit-Aktionen

- werden immer ausgeführt beim Verlassen des Zustandes

interne Aktion

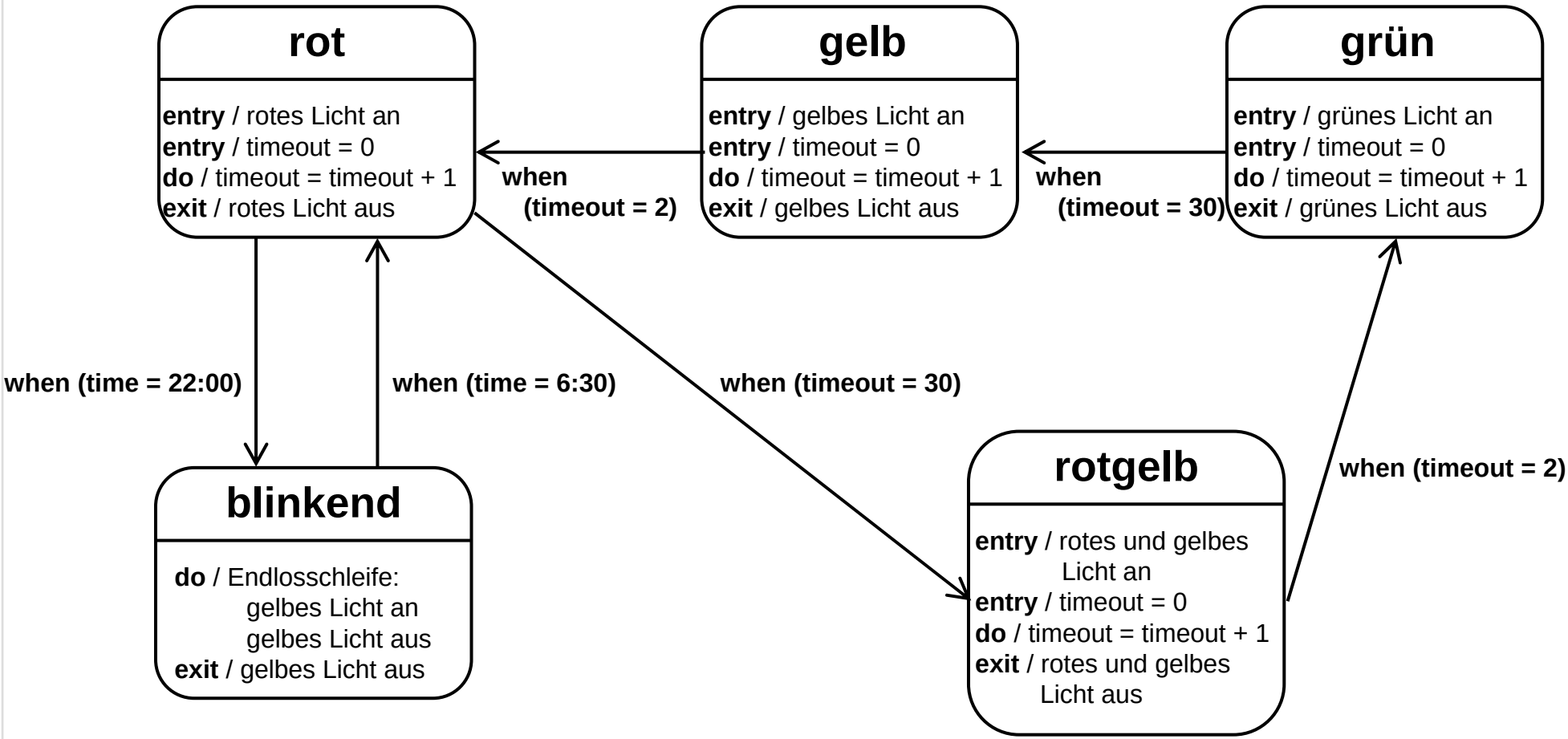
- Aktion, die durch irgendein anderes Ereignis ausgelöst wird, aber nicht den Zustand verlässt.

do-Aktivität

- (längere) Aktion, die ausgeführt wird, während der Zustand andauert
- wird entweder beendet durch vorgesehene Beendigung der Aktion oder durch Auftreten eines Ereignisses, das einen Zustandsübergang verursacht
- entspricht einer internen Aktion, die unmittelbar ausgeführt wird (also ohne spezielles Ereignis)

UML: Zustandsübergangsdiagramme (Statecharts)

Beispiel Verkehrsampel:



Achtung: Ein Detail hier ist schlecht spezifiziert !

UML: Aktivitätsdiagramme

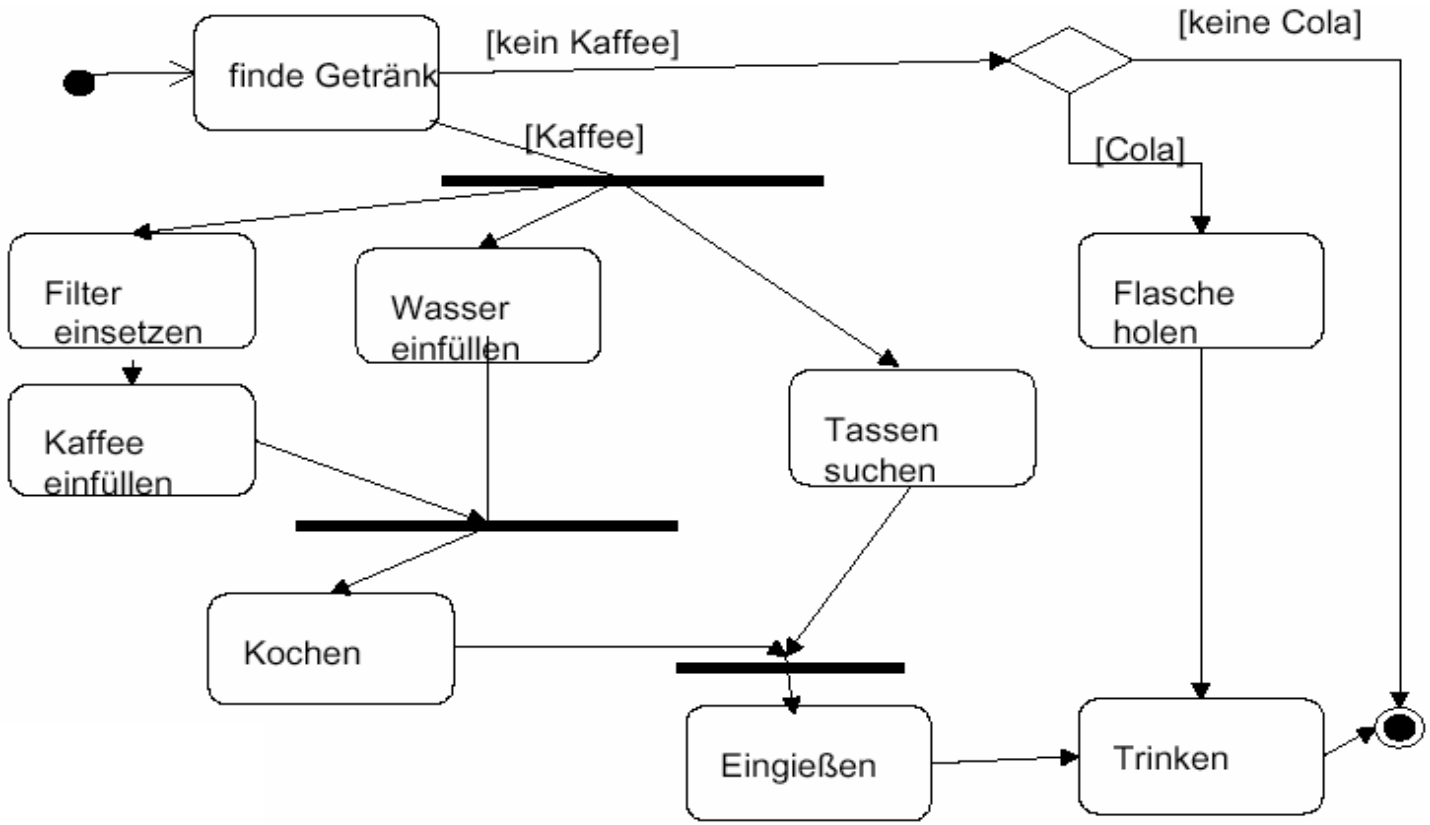
zur Beschreibung von Daten-/Prozess-Zusammenhängen

für spezielle Zustandsübergänge:

Transitionen werden grundsätzlich durch die Beendigung von Aktionen ausgelöst

spezielle Notation für Parallelität und Synchronisation

Beispiel:



Zusammenfassung: UML

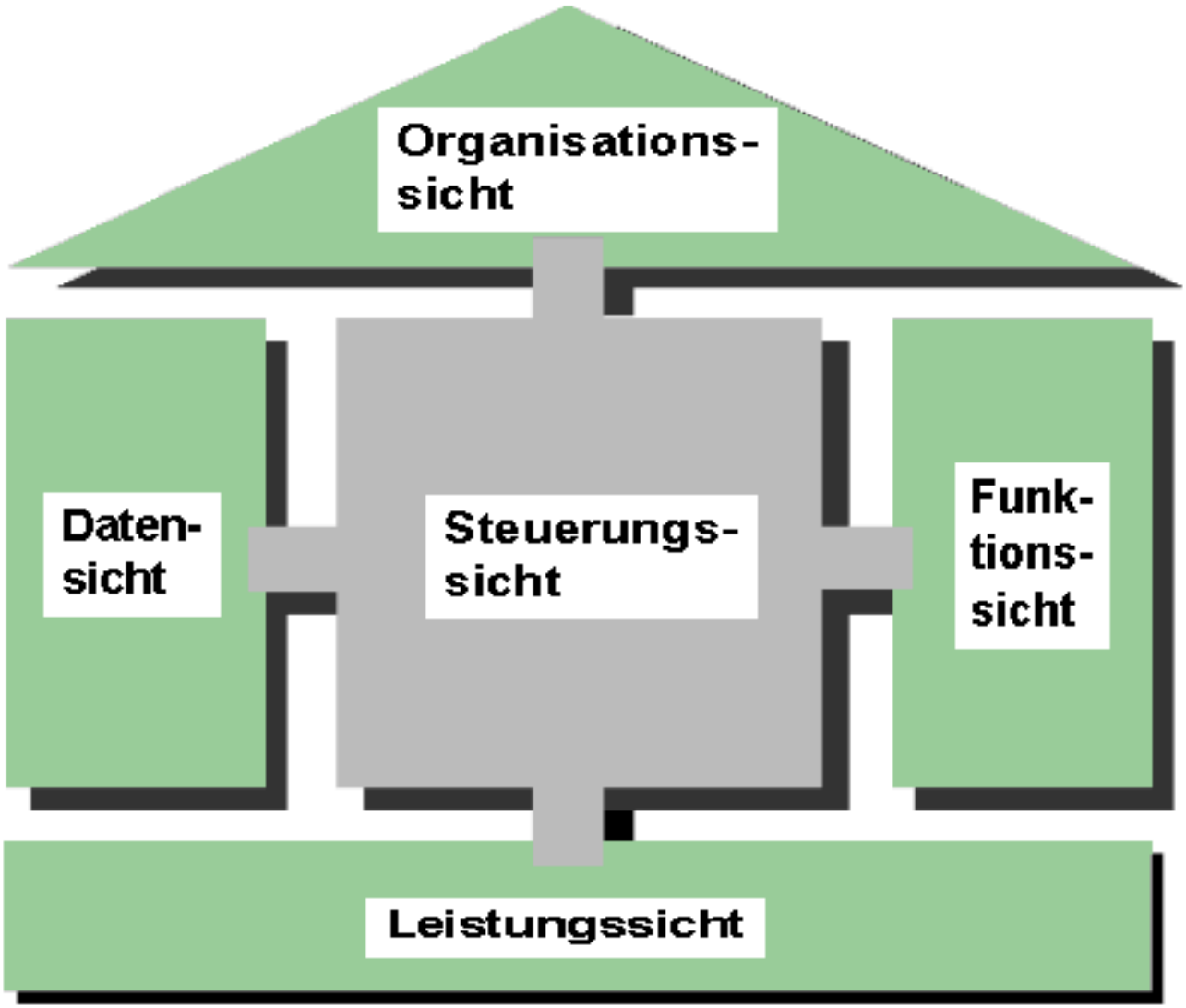
- objektorientierte Systembeschreibung: Klassendiagramme mit Beziehungen und Hierarchien
- Beschreibung von Anwendungsszenarien (Use Cases)
- Beschreibung von Prozessabfolgen (Sequenzdiagramme)
- Beschreibung von Zustandsabfolgen (Zustandsübergangsdiagramme)
- Aktivitätsdiagramme: spezielle Zustands-Aktivitäts-Beschreibungen
- *Kooperationsdiagramme*
- *Implmentierungsdiagramme*
- *Paketdiagramme*

Literaturempfehlung:

Jochen Seemann / Jürgen Wolff von Gudenberg: *Software-Entwurf mit UML*
Springer-Verlag 2000, ISBN 3-540-64103-3
(vergriffen, Neuauflage für Juli 2005 angekündigt)

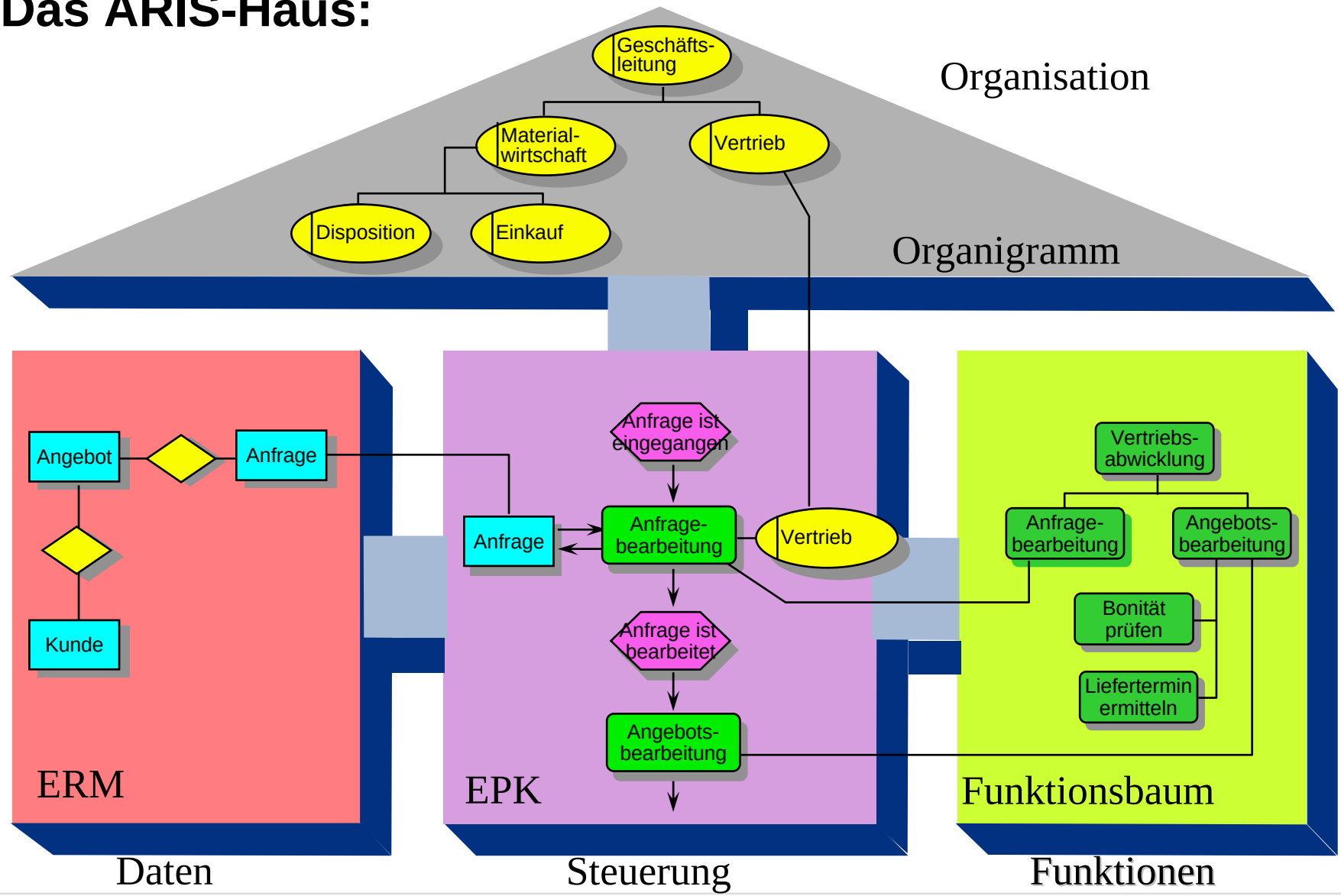
ARIS: Architektur integrierter Informationssysteme

Das ARIS-Haus:



ARIS: Architektur integrierter Informationssysteme

Das ARIS-Haus:



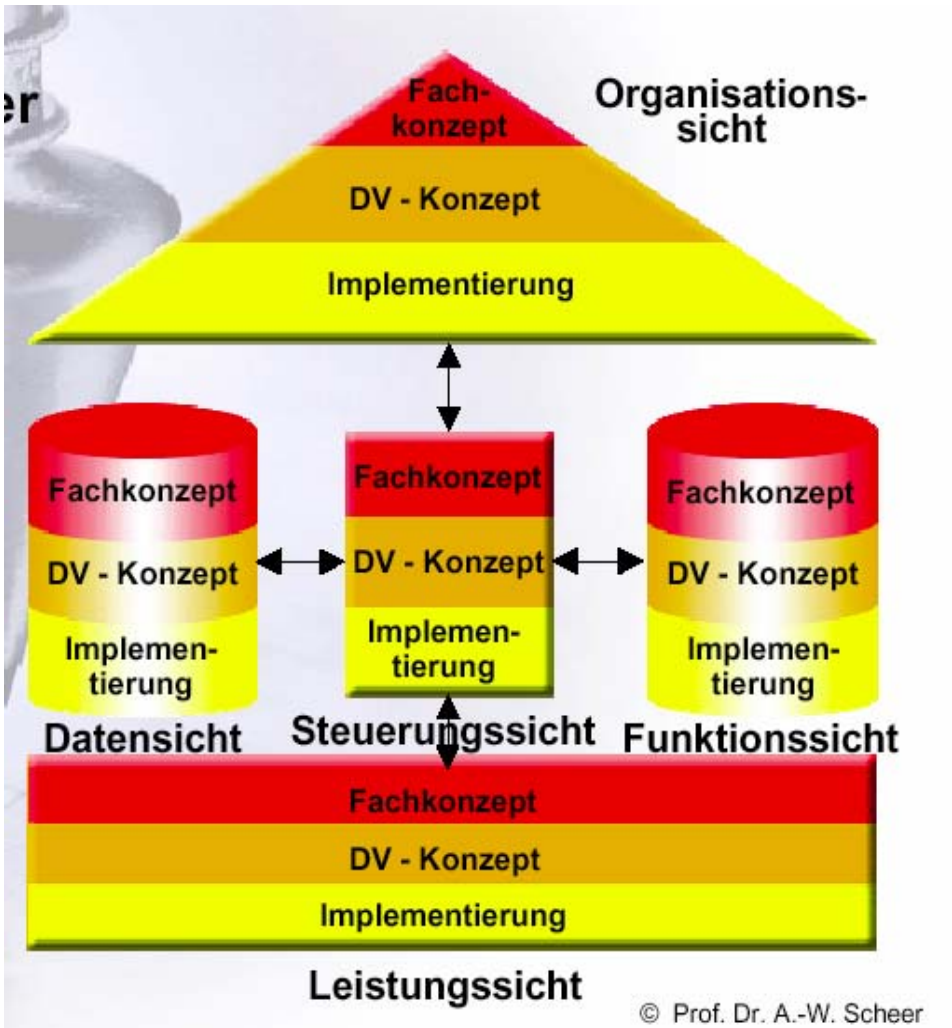
ARIS: Architektur integrierter Informationssysteme

Die Denkwelt von ARIS-Benutzern:

in der Regel aus betriebswirtschaftlicher Sicht

ARIS: Architektur integrierter Informationssysteme

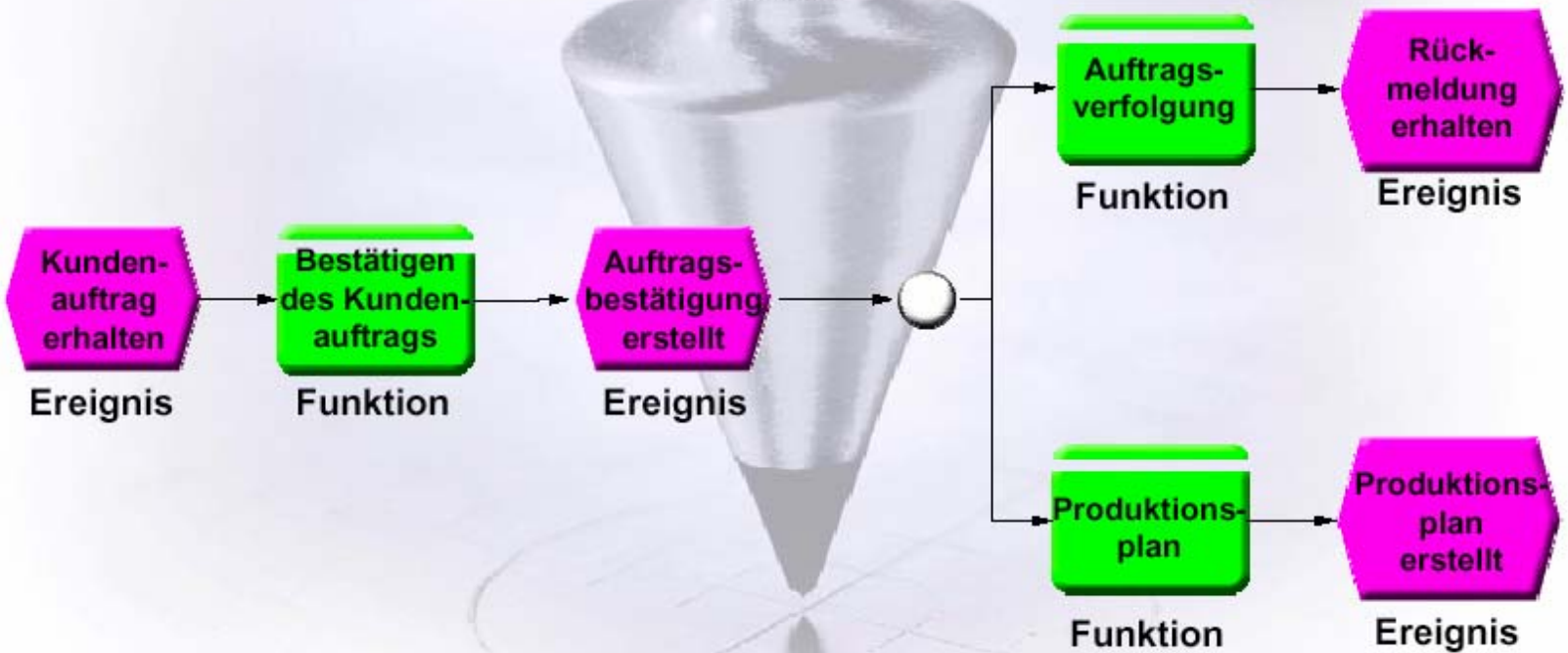
Die Denkwelt von ARIS-Benutzern:



ARIS: Ereignisorientierte Systemmodellierung

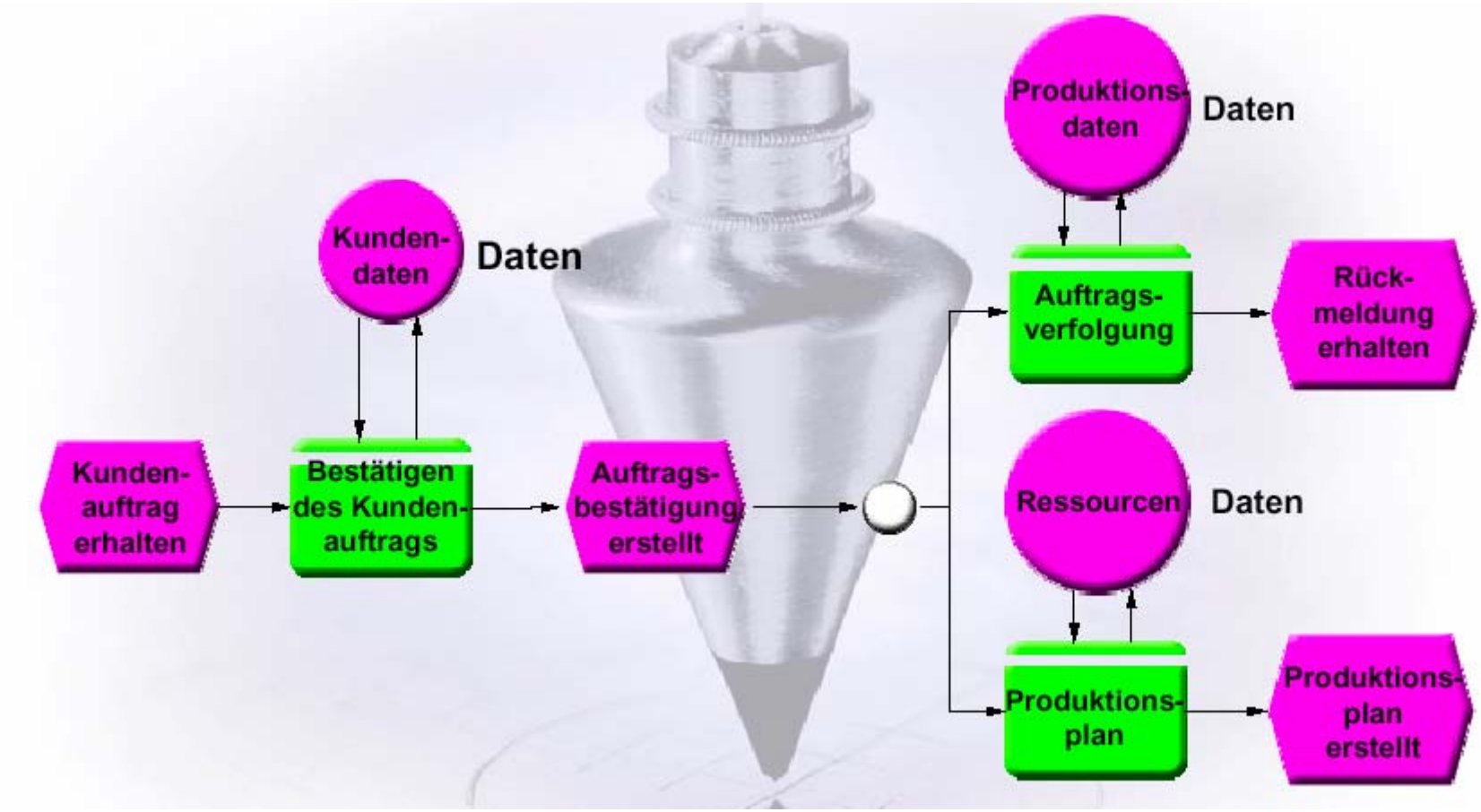
Ereignisse lösen Funktionen aus

Funktionen generieren Ereignisse



ARIS: Ereignisorientierte Systemmodellierung

Verarbeitung von Daten in *Prozessen* (Funktionen)

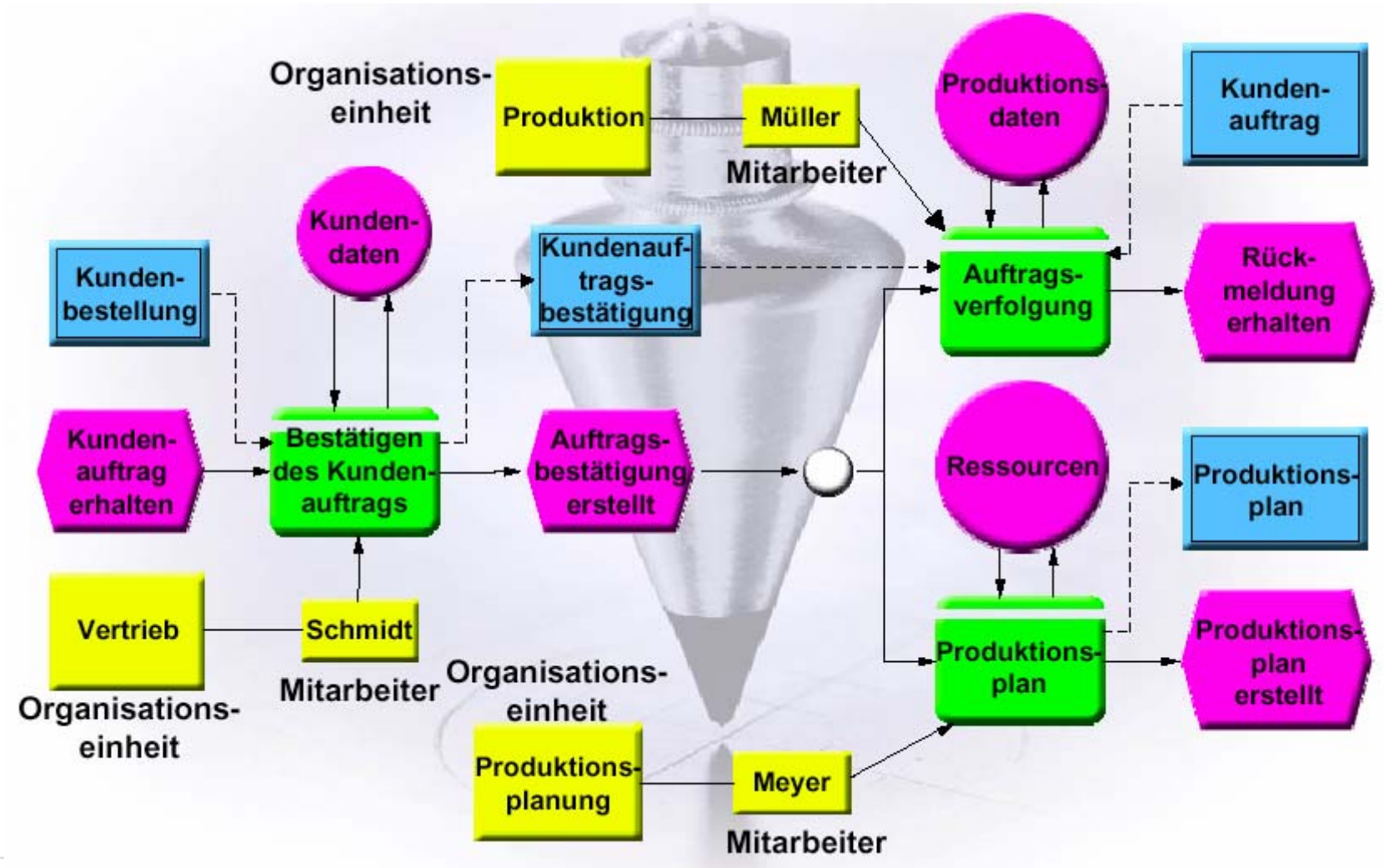


ARIS: Ereignisorientierte Systemmodellierung

Modellierung der *Verantwortlichkeiten*:

ARIS: Ereignisorientierte Systemmodellierung

Modellierung von *Leistungen*:



Beim nächsten Mal:
EPKs und weitere ARIS-Details
Zusammenfassung Systemanalyse / Softwareentwurf