

Software Engineering 1

Hochschule für angewandte Wissenschaften Hamburg

Fachbereich Informatik

Prof. Dr. Stefan Sarstedt

(stefan.sarstedt@haw-hamburg.de)

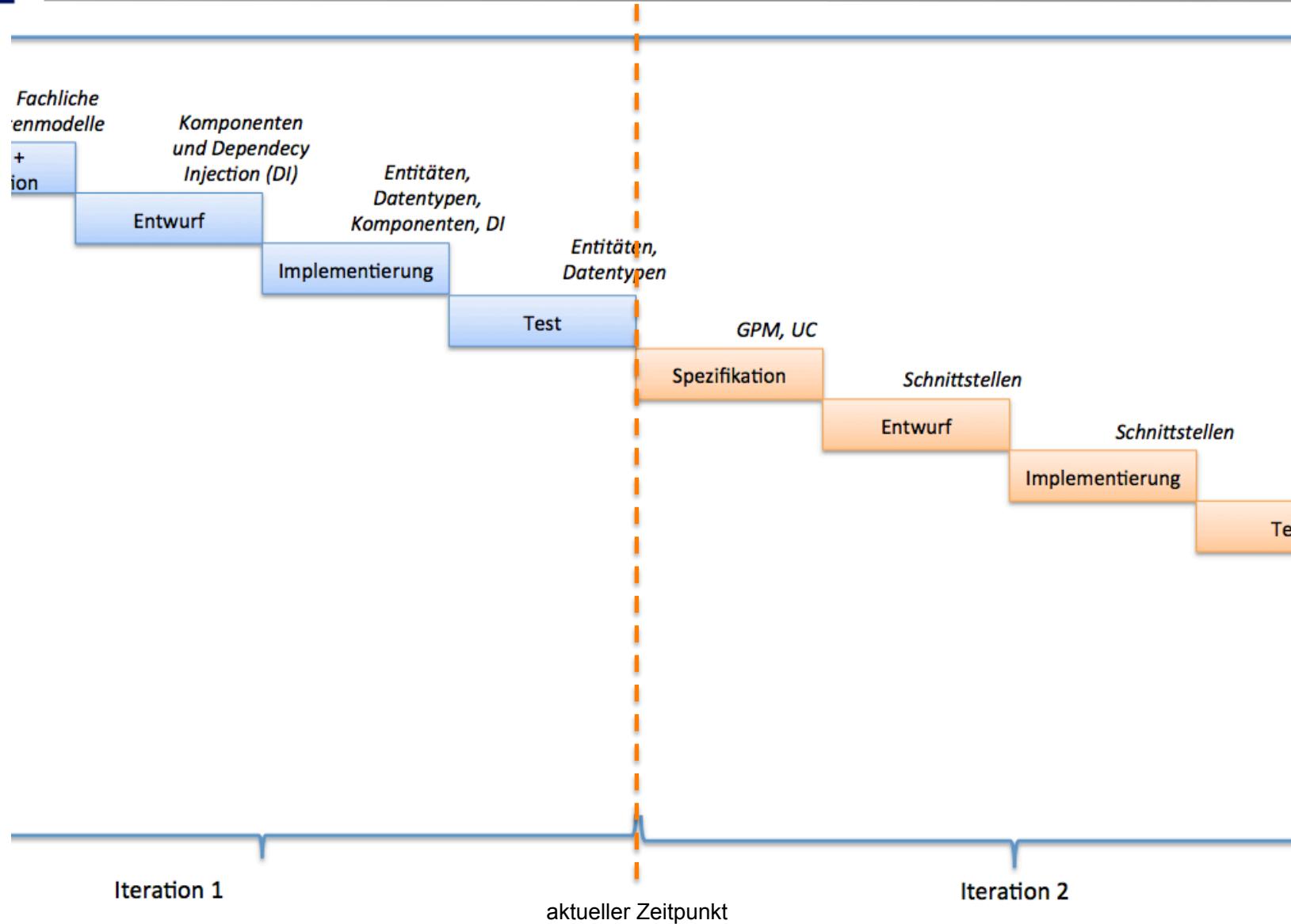
Raum: 10.85

Projektablauf

Analyse

Spezifikation

Iteration 2



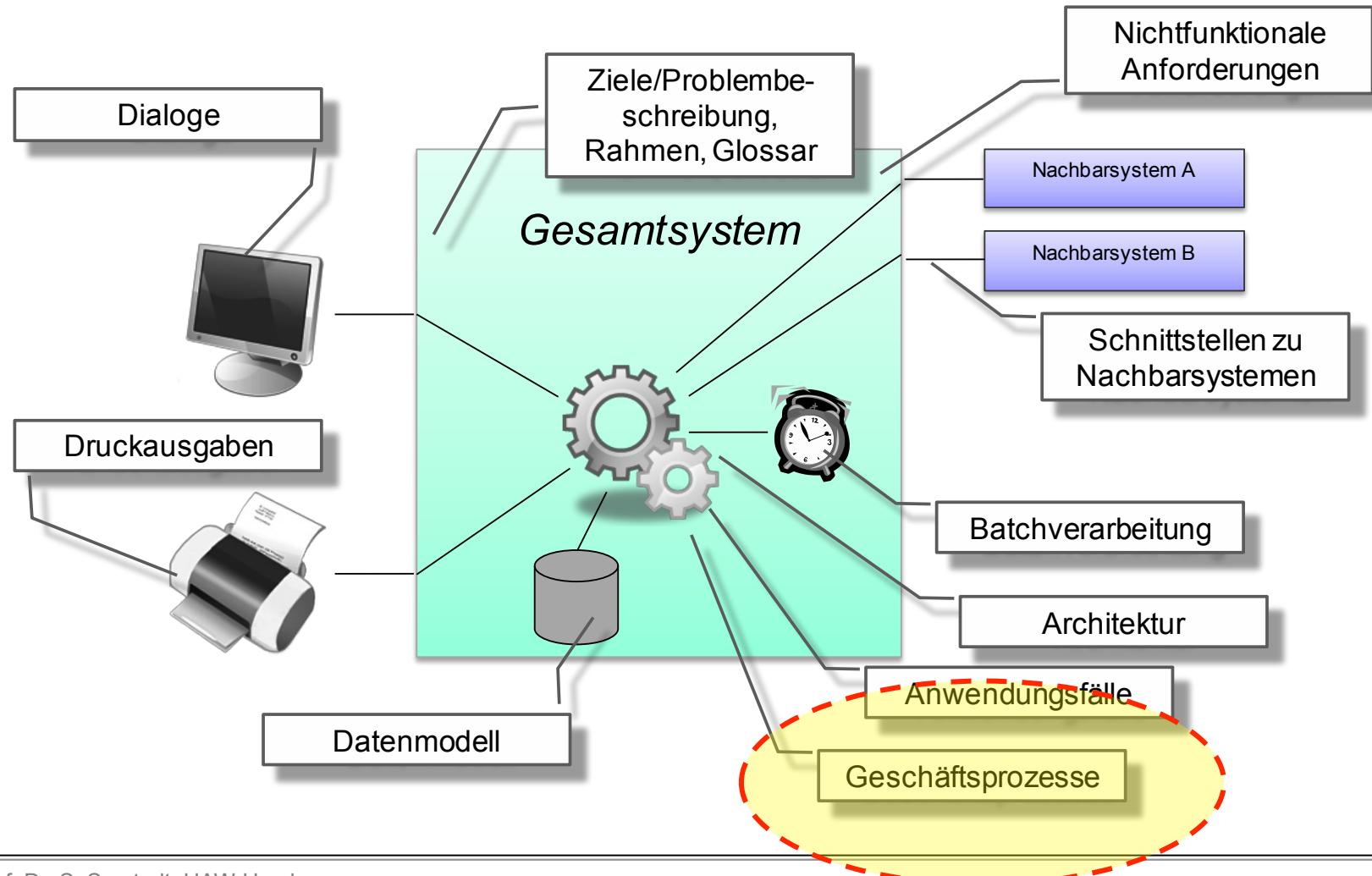


Funktionalität des Systems

- Wir „kennen“ nun die „Daten“, „Begriffe“ aus dem Anwendungsbereich und deren Zusammenhänge (Fachliches Datenmodell, Entitäten, Datentypen, etc.!)
- Aber was sind die **Abläufe**?
- Und welche Teile dieser Abläufe sollen **durch unser zu erststellendes System** unterstützt werden?

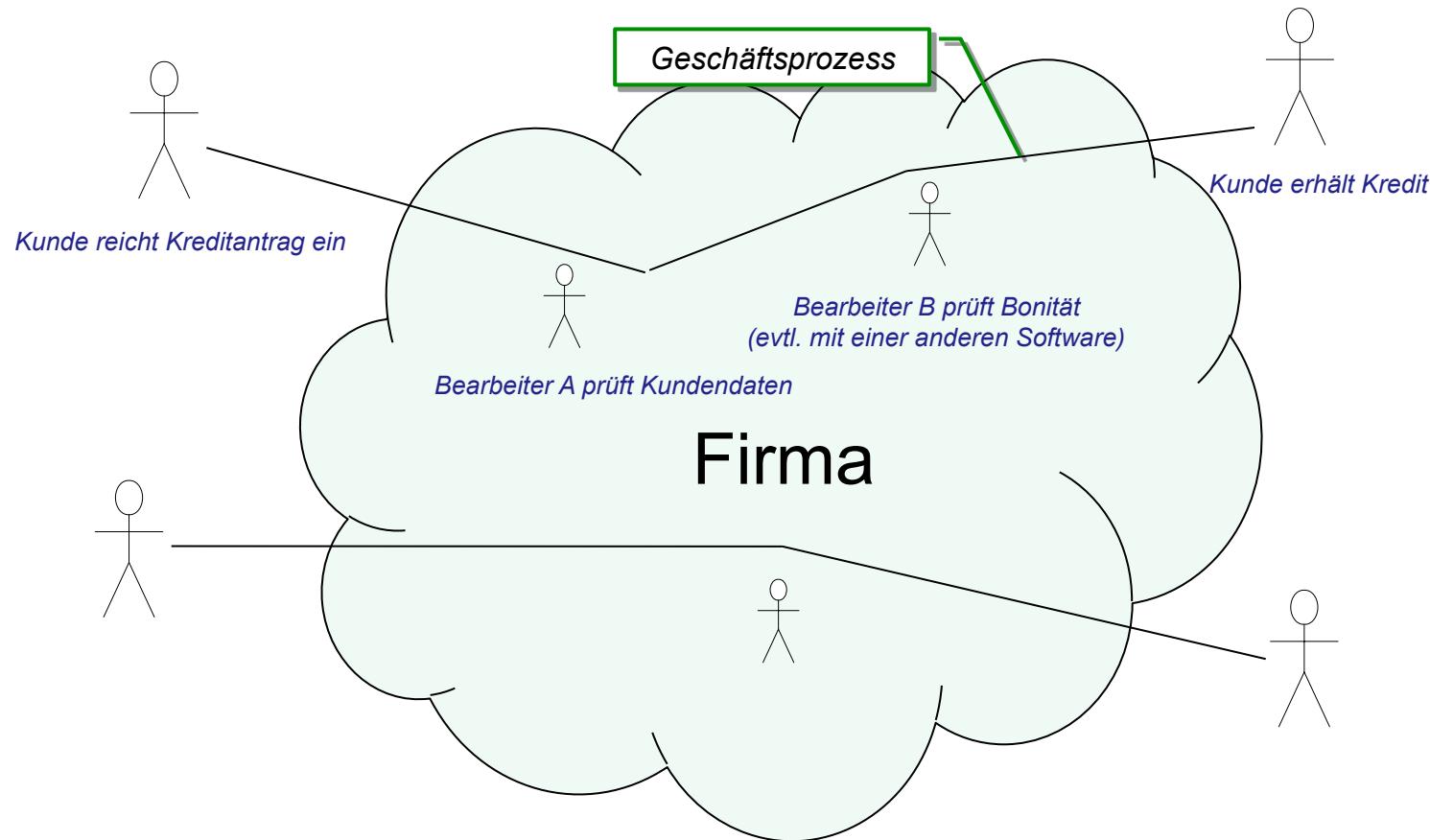


Spezifikation – Geschäftsprozesse





Geschäftsprozesse





Was ist ein Geschäftsprozess?

*„Ein Geschäftsprozess (GP) besteht aus einer Menge logisch verknüpfter Einzeltätigkeiten (Aufgaben, Aktivitäten), die ausgeführt werden, um ein geschäftliches oder **betriebliches Ziel** zu erreichen.“*

*Ein Geschäftsprozess ist **wiederholbar**, mit Wertschöpfung verbunden und nutzt die Ressourcen einer oder mehrerer Organisationen. Er kann Teil eines anderen Geschäftsprozesses sein oder andere Geschäftsprozesse enthalten bzw. diese anstoßen.*

*Geschäftsprozesse gehen **oft über Abteilungs- und Betriebsgrenzen hinweg** und gehören zur Ablauforganisation eines Betriebs.“*

<http://de.wikipedia.org/w/index.php?title=Gesch%C3%A4ftsprozess&oldid=134674893>



Geschäftsprozesse (1/2)

Ein Geschäftsprozess zeichnet sich aus durch:

- einen definierten Anfang/definiertes Ende,
- erforderliche Eingaben (z.B. Kundenwünsche),
- produzierte Ergebnisse (z.B. Befriedigung der Kundenwünsche)

Jeder Geschäftsprozess

- hat ein (Teil-) Ziel des Unternehmens zu erfüllen

Alle Geschäftsprozesse zusammen

- erfüllen das Gesamtziel des Unternehmens

Konsequenz:

- Gestaltung und Entwurf von geschäftsprozessrealisierenden IT-Systemen müssen sich an den Geschäftsprozessen und den Unternehmenszielen orientieren.



Geschäftsprozesse (2/2)

- Ein **Geschäftsprozess ist eine Folge von Aktivitäten**, die
 - in einem logischen Zusammenhang stehen,
 - inhaltlich abgeschlossen sind und
 - unter Zuhilfenahme von Ressourcen
 - und eingehenden Informationen
 - durch Menschen und/ oder Maschinen
- auf ein Unternehmensziel hin ausgeführt werden.
- Synonyme für Geschäftsprozess:
 - Business Process
 - Unternehmensprozess
 - betrieblicher Ablauf
 - Vorgang



Beispiele für Geschäftsprozesse

- Bearbeitung eines Schadensfalls in einer Versicherung
- Kreditüberprüfung in einer Bank
- Ausschreibung, Vergabe von Bauprojekten in der Stadtverwaltung
- Reklamationsprozesse in einem Versandhaus
- Wartung technischer Geräte in einem Kraftwerk
- Entwicklung von Software in einem Systemhaus
- Just-in-Time Logistik eines Automobilzulieferers
- Planung, Buchung und Abrechnung einer Reise im Reisebüro
- Beantragung und Abrechnung einer Dienstreise
- Einstellung und Entlassung von Angestellten / Personal



Beispiele für **Aktivitäten**

- Angebot erstellen
- Angebot akzeptieren
- Auftrag bearbeiten
- Einen Spediteur mit einem Transport beauftragen
- Angebot manuell archivieren
- Kinokarten am Automaten bezahlen
- Eine Lerngruppe einrichten
- Im Hotel einchecken
- Ein Konto eröffnen
- ...
- **Aktivitätskategorien:** strukturiert, atomar, manuell, automatisch, transaktional, ...

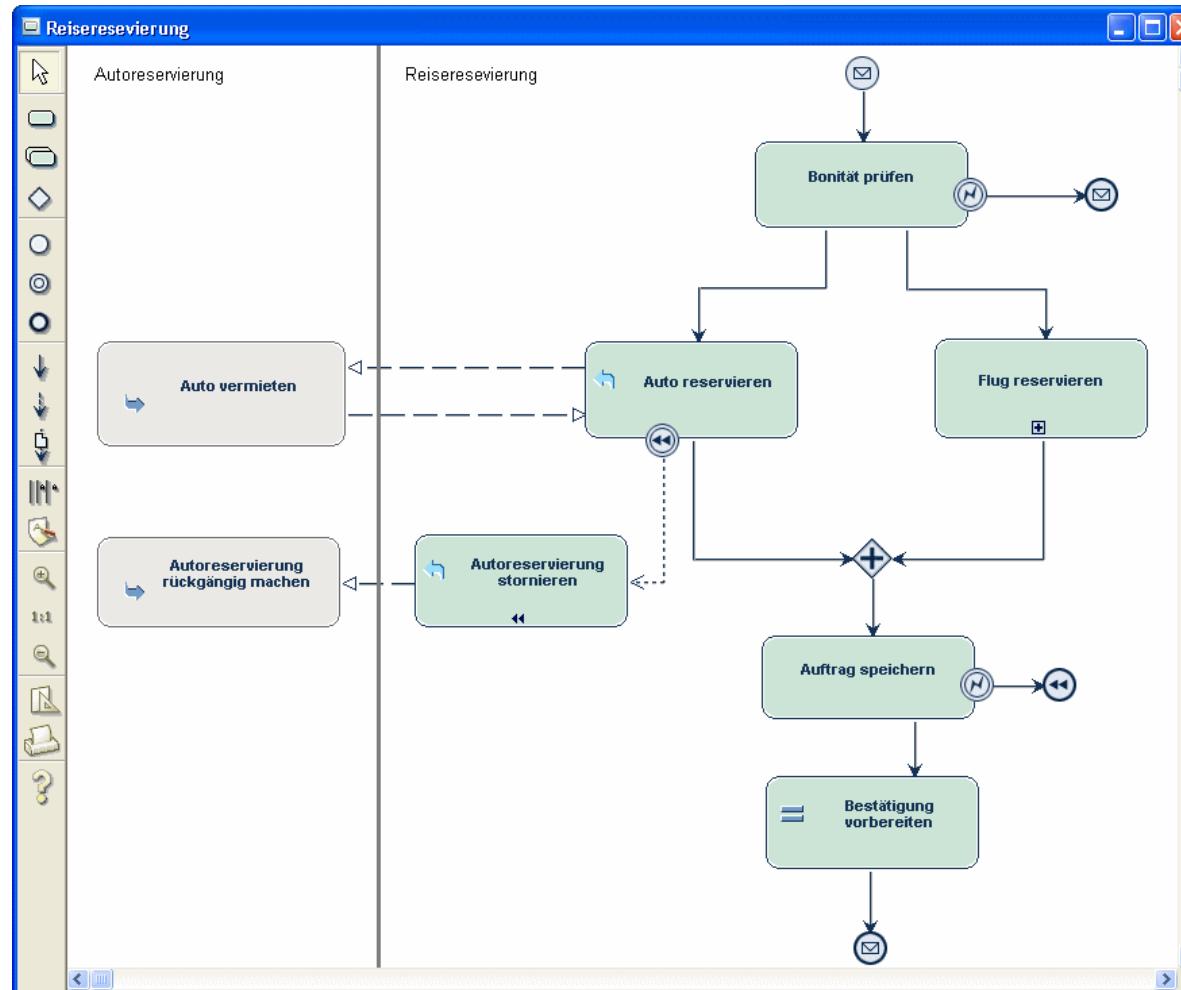


Geschäftsprozessmodellierung (GPM)

- Ziele in der **Geschäftsprozessmodellierung** (GPM) in der Betriebswirtschaft: Dokumentation, Produktivitätssteigerung, Reorganisation, Controlling, ...
- **Ziele im Software Engineering:** Verständnis über und Dokumentation der betrieblichen Abläufe (IST) und Festlegung von neuen, hoffentlich besseren, Abläufen (SOLL)

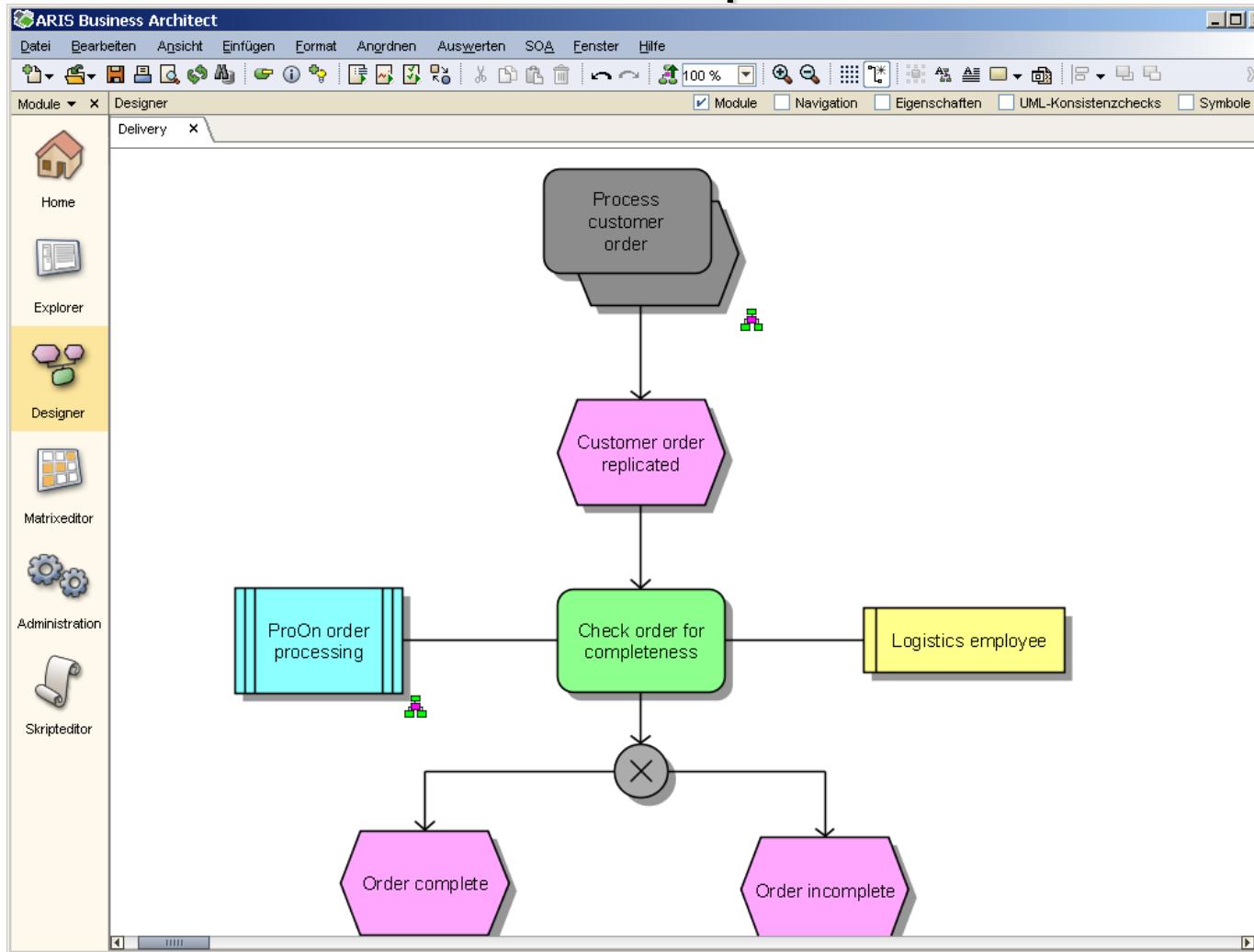


GPM – Beispiele



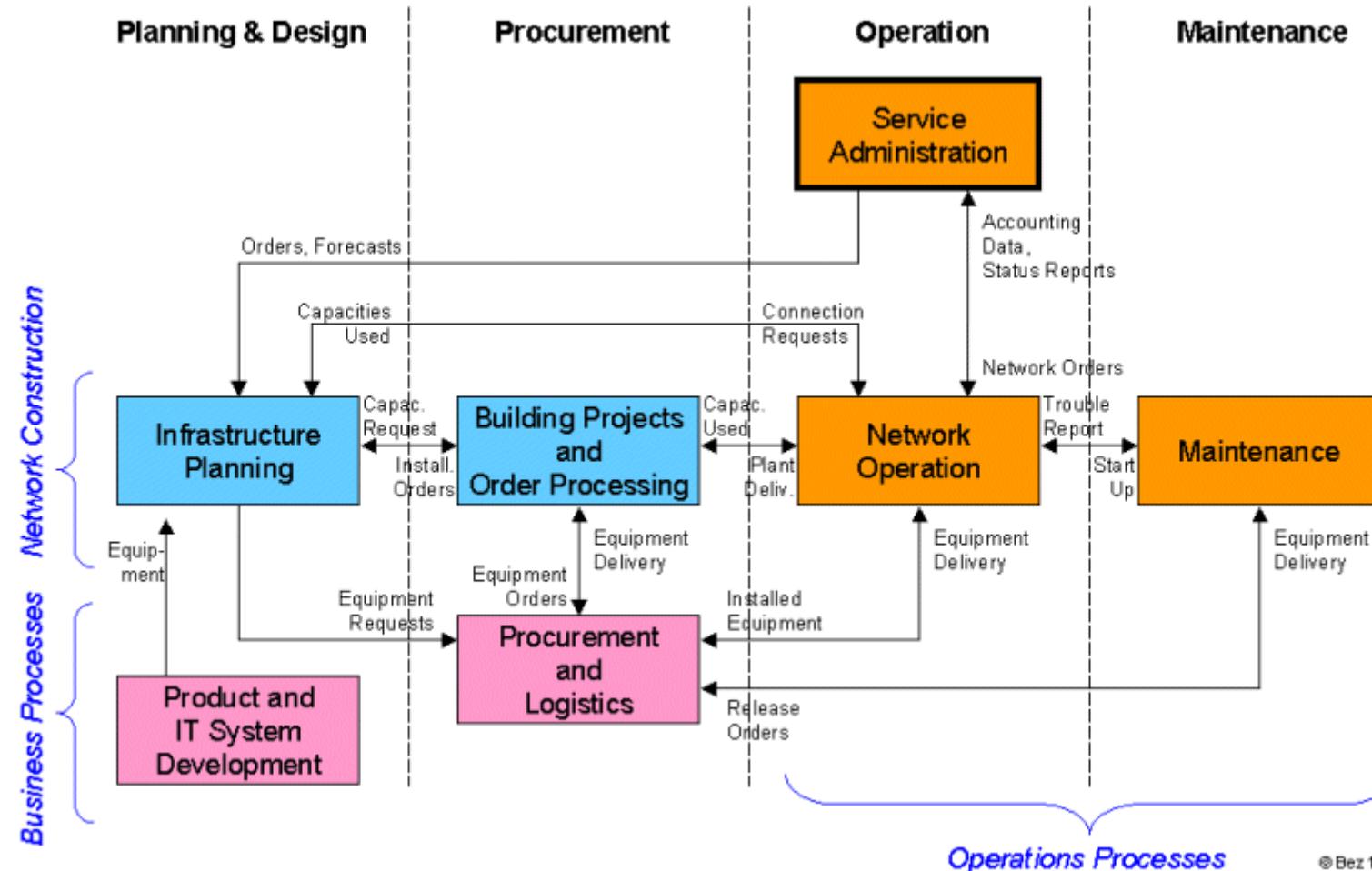


GPM – Beispiele



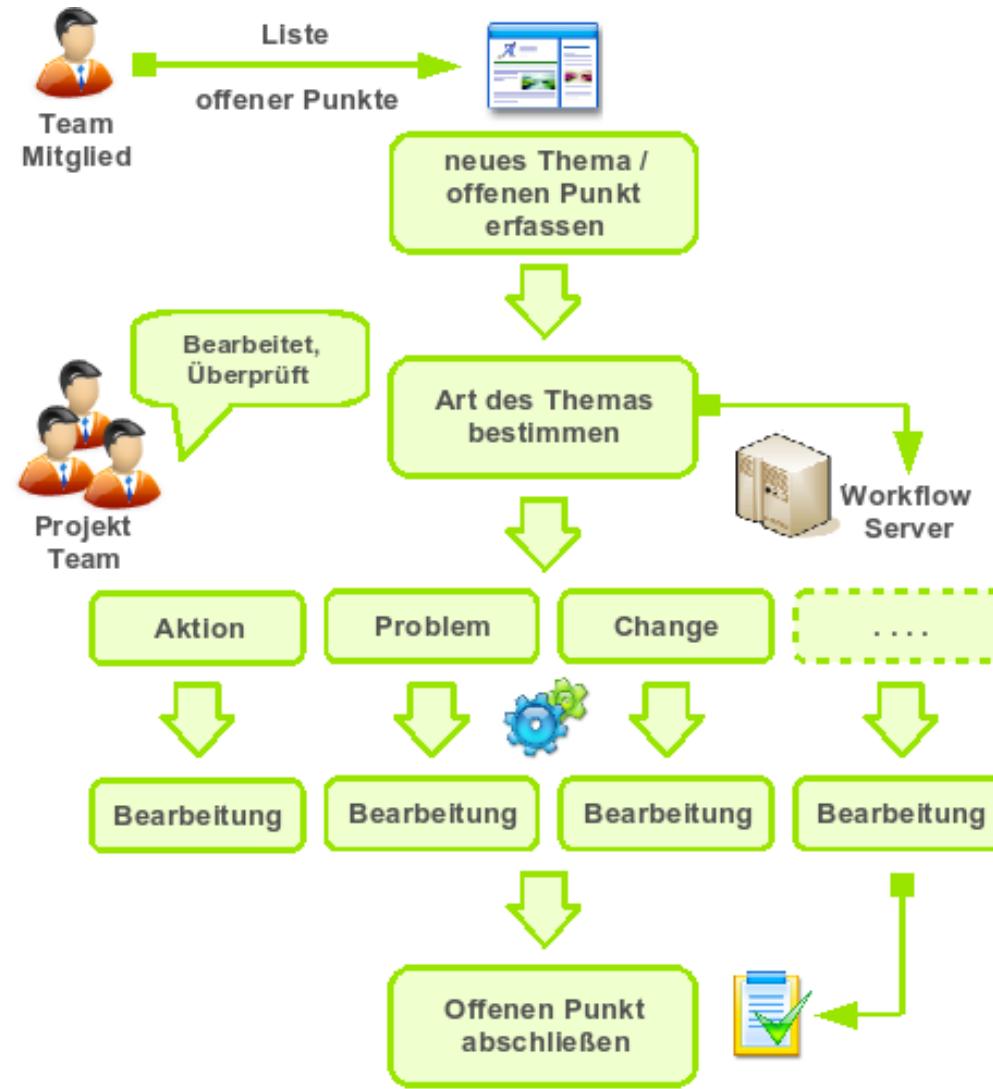


GPM – Beispiele



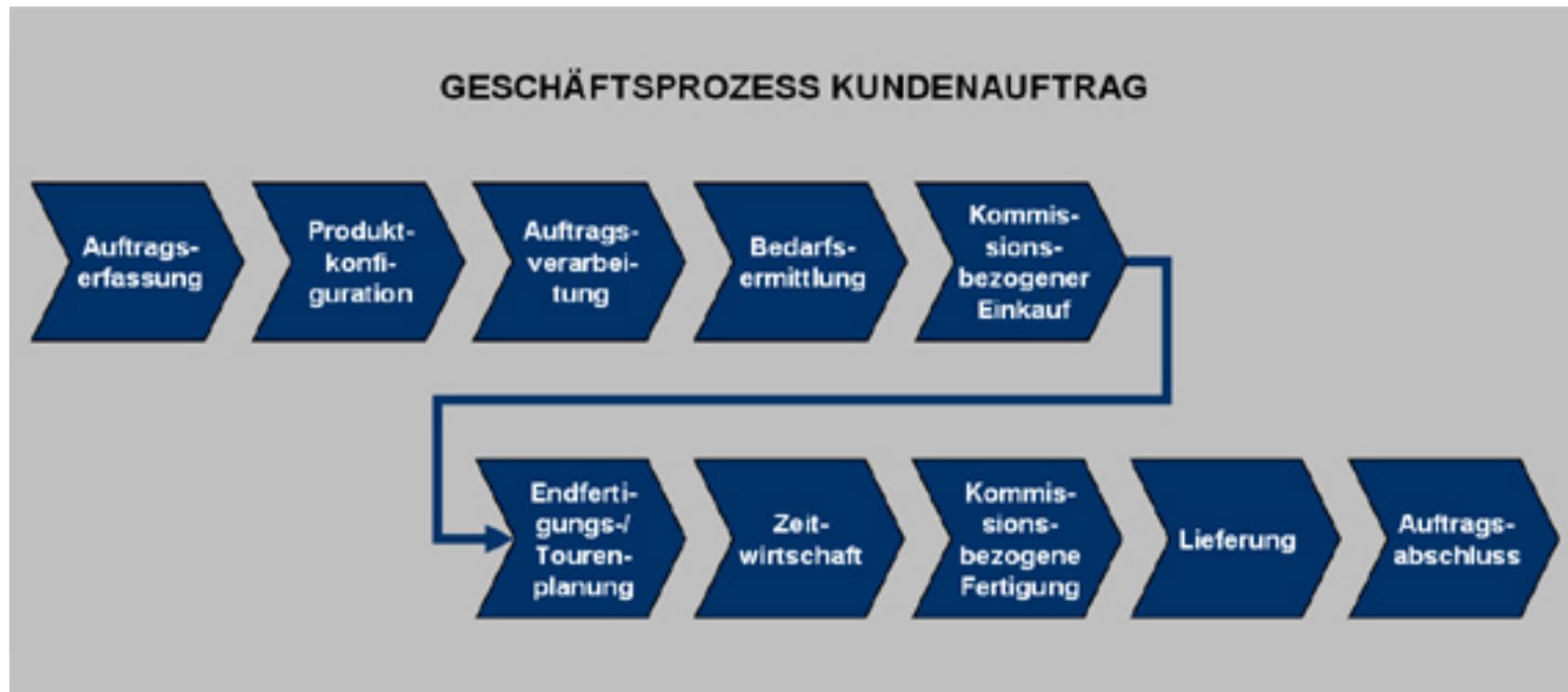


GPM – Beispiele





GPM – Beispiele





Aufgabe: Geschäftsprozessmodellierung (GPM)

Geschäftsprozess

Formulieren Sie einen möglichen Geschäftsprozess „Ins Kino gehen“ (von der Entscheidung bis zum Ansehen des Films) in eigener Notation.

Stellen Sie die verschiedenen Verantwortlichkeitsbereiche („Kunde“, „Mitarbeiter an der Kasse“, etc.), mit deren Tätigkeiten, dar.

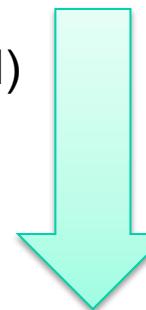


20 Minuten



Geschäftsprozessmodellierung – Notationen

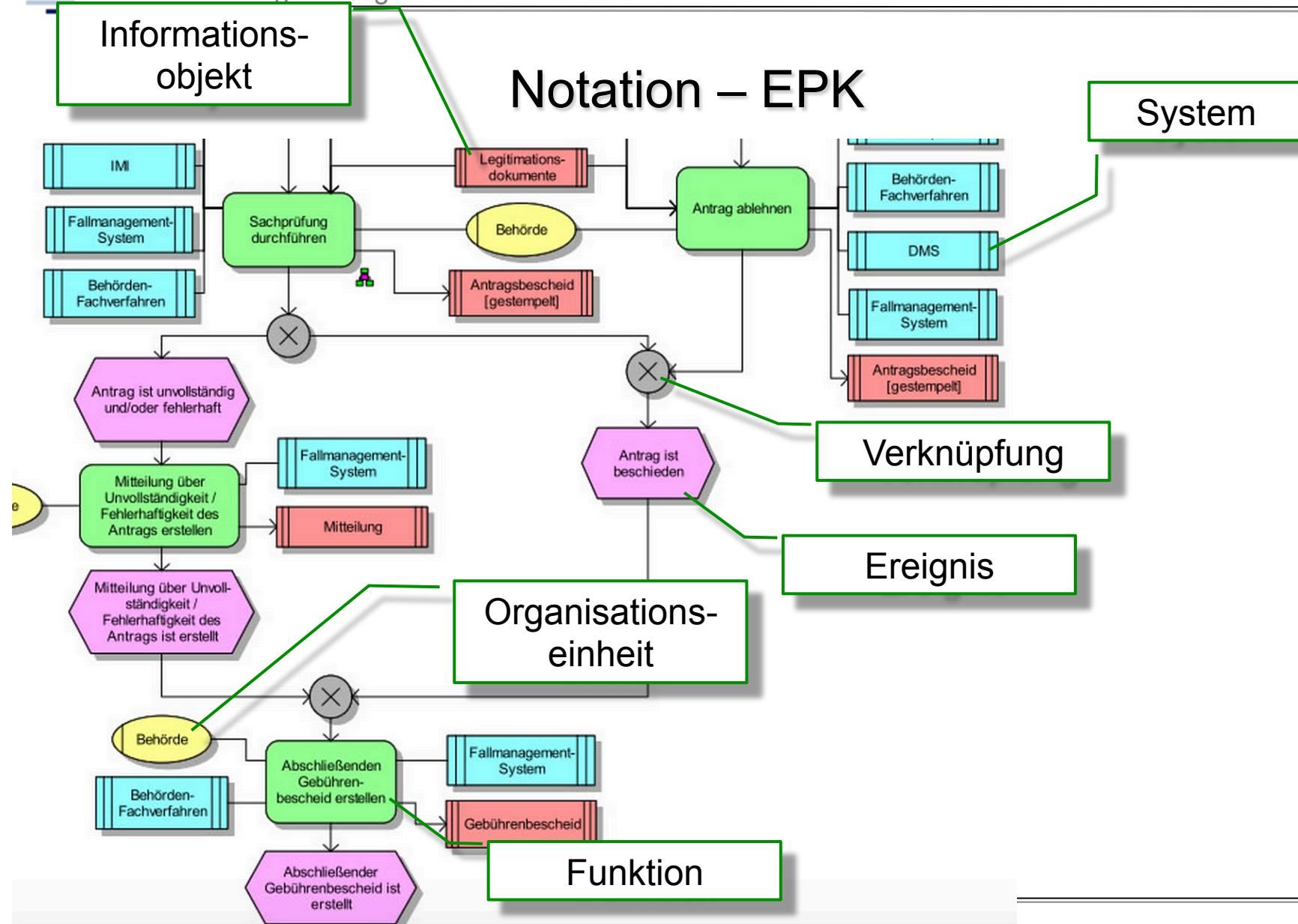
- Die vorherigen Beispiele verwenden zum großen Teil „eigene“ Notationen. Nachteile:
 - Verständnis?
 - Ausdrucksmächtigkeit?
- Daher gibt es auch hier **Standards**
 - Business Process Modeling Notation (BPMN)
 - UML Aktivitätsdiagramme
 - Ereignisgesteuerte Prozessketten (EPK)
 - ...



„aktuell sehr relevant“

„aktuell weniger relevant“

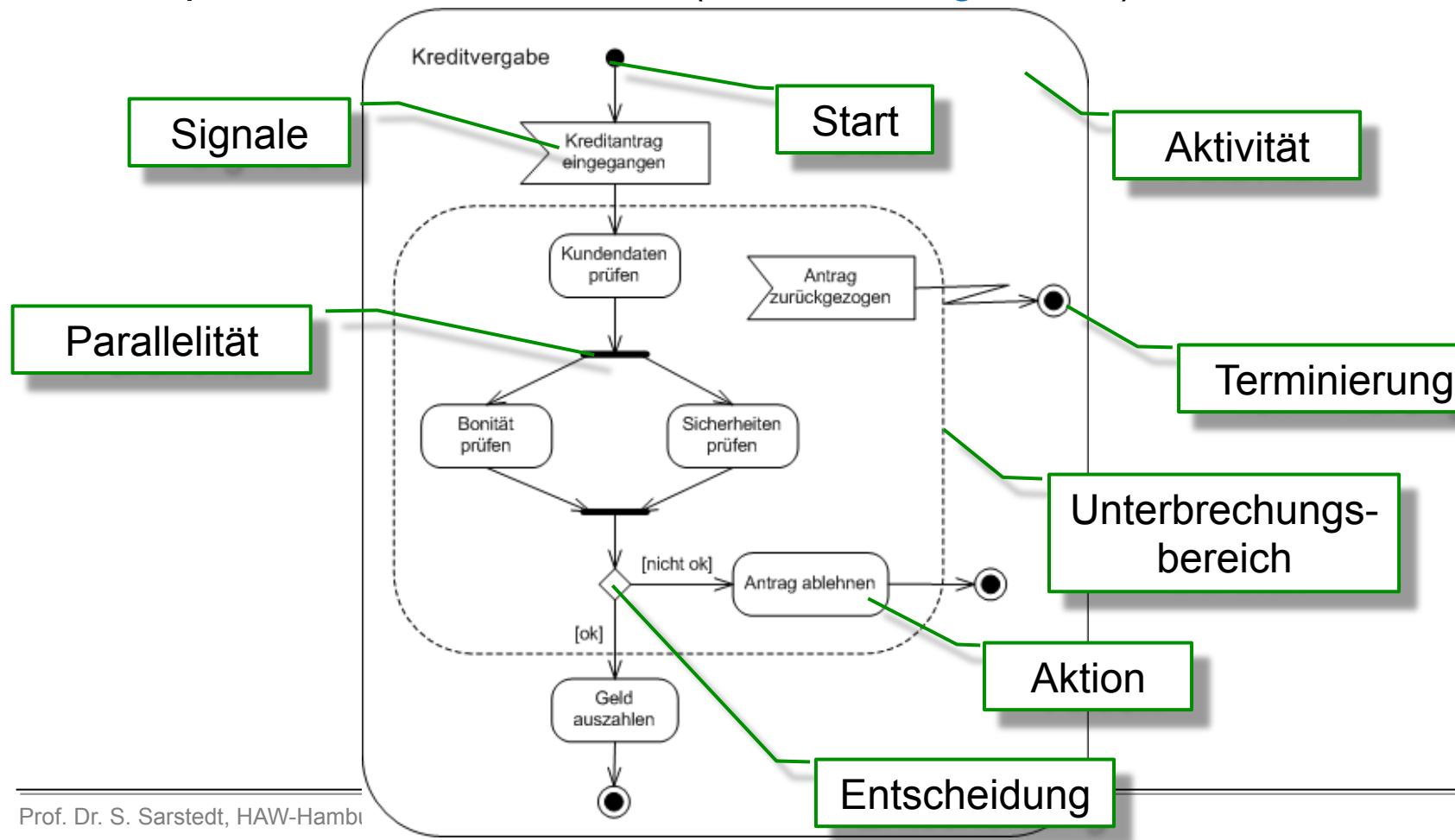
Notation – EPK





Notation – UML Aktivitätsdiagramme

- Beispiel in UML 2 – Notation (Aktivitätsdiagramme)



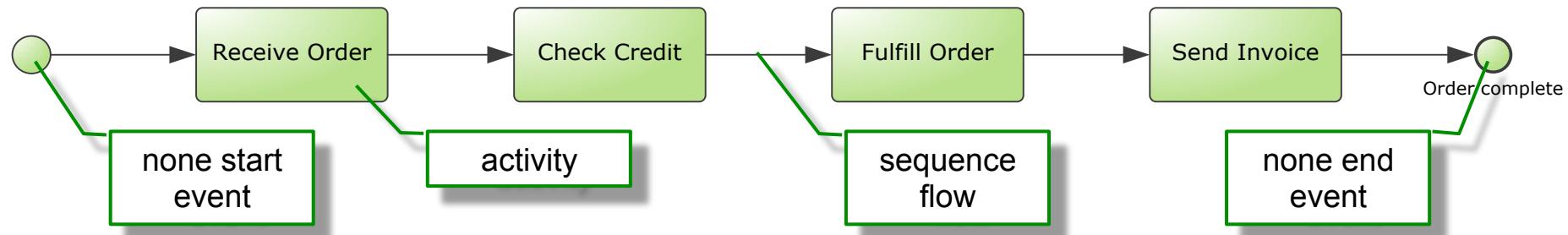


BPMN – Business Process Modeling Notation

- BPMN ist eine (mächtige) Sprache zur Modellierung von Geschäftsprozessen
- Standardisiert durch die Object Management Group ([OMG](#))
 - Aktuelle Version der Spezifikation: 2.0: <http://www.bpmn.org/>
- Im Folgenden nur eine oberflächliche Einführung!



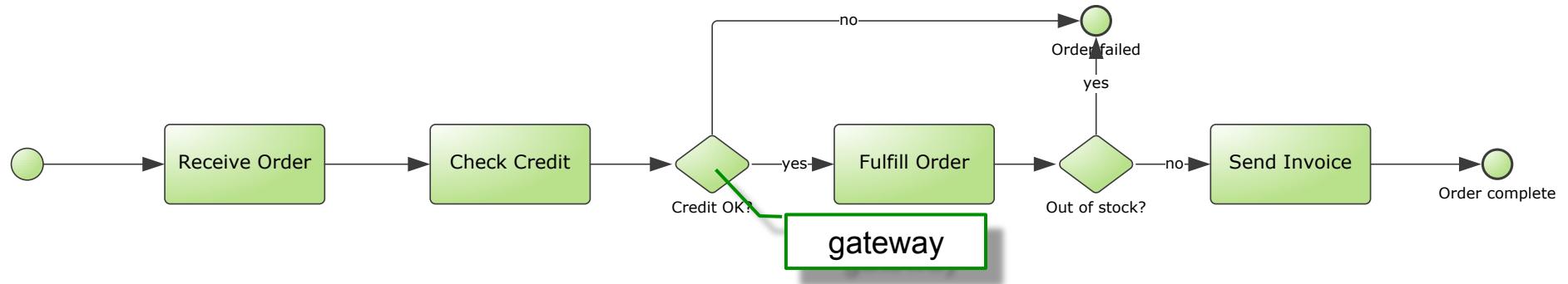
Notation – *activities* und *flows*



- Das Diagramm beschreibt nicht den kompletten Verkaufsprozess
- Es beschreibt den „normalen Ablauf“, den sogenannten „happy path“ 😊
- Es fehlen z.B. Ausnahmefälle
 - „Kredit nicht ausreichend“
 - „Lager leer“
 - ...



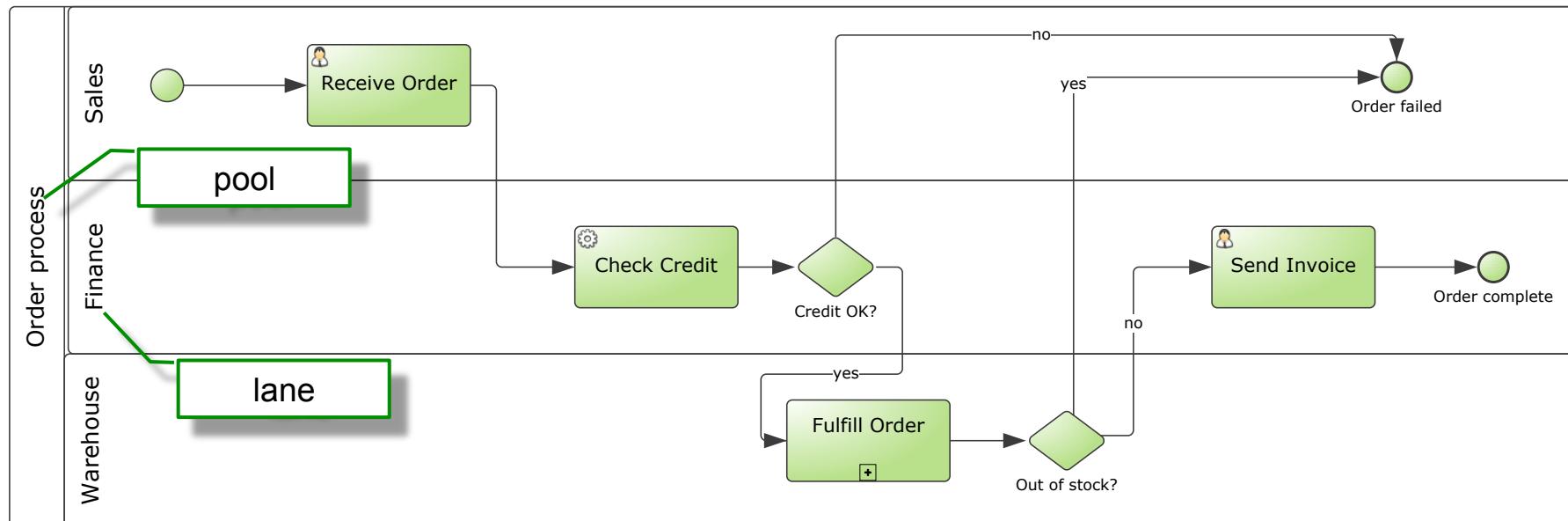
Notation – gateways



- *gateways verzweigen den Fluss*
 - Es gibt mehrere Arten von *gateways* → später
 - alternative Darstellung (geläufig in BPMN 1.x):
- mehrere *end events* repräsentieren unterschiedliche Endzustände



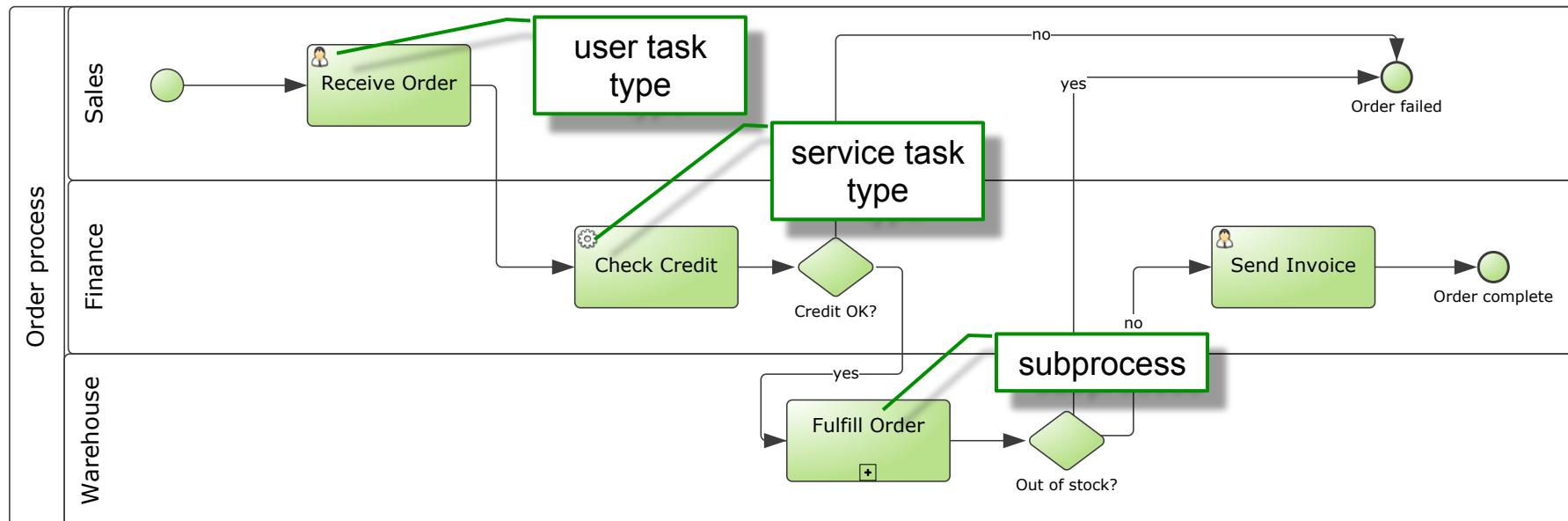
Notation – lanes



- *lanes* repräsentieren **Rollen** oder **Organisationseinheiten**
 - wichtig nur für *activities*
 - es ist egal, wo *start events*, *end events*, *gateways*, usw.
- Ein *pool* repräsentiert einen Prozess
 - mehrere Pools können miteinander interagieren → später



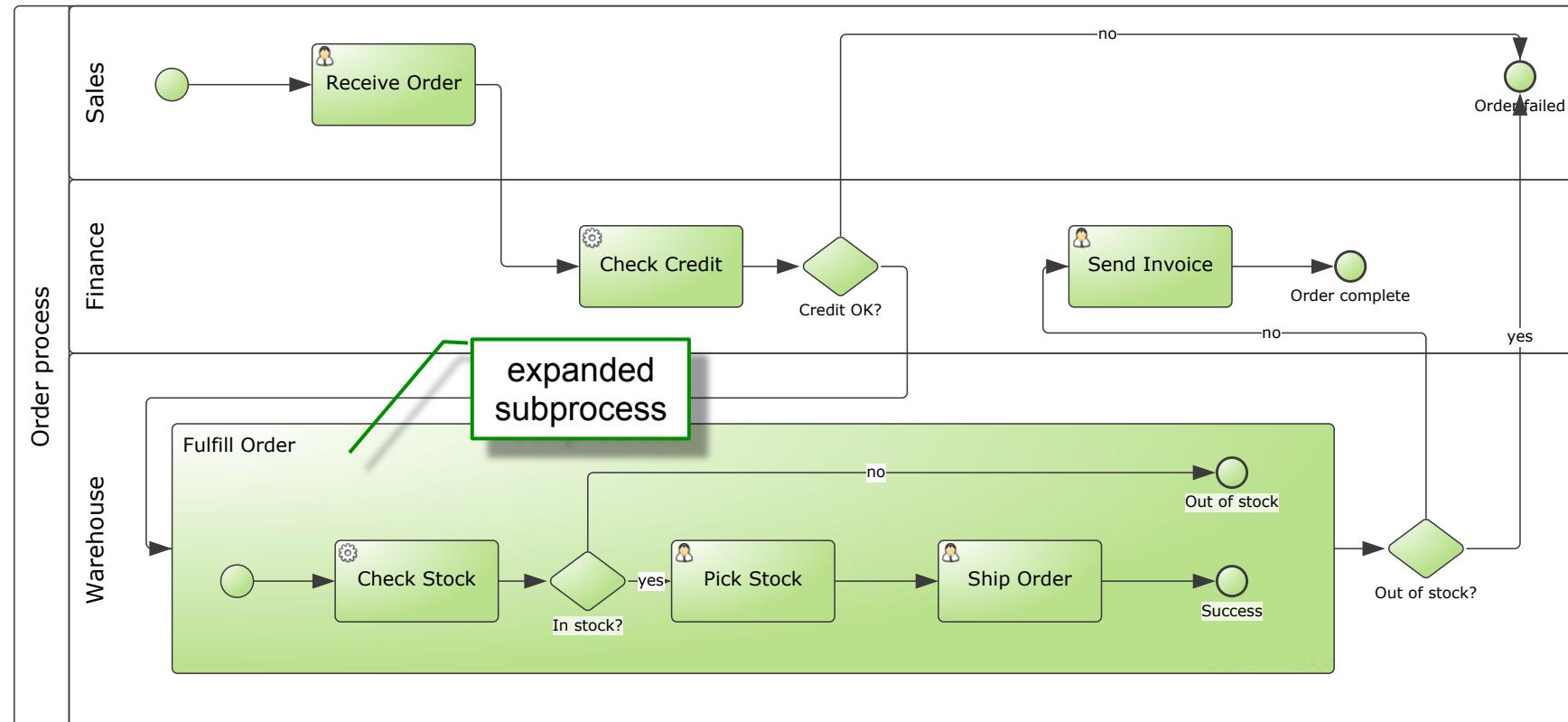
Notation – *task types* und *subprocesses*



- *tasks* sind atomare Aktivitäten; sie stellen **Aktionen** dar und sollten deshalb im Stil **VERB+NOMEN** benannt werden
- *task types* unterscheiden z.B. Benutzeraktivitäten von automatischen Aktivitäten (Typ „none“: undefiniert)
- Durch *subprocesses* ist eine Hierarchiesierung möglich
 - der *subprocess* „Fulfill Order“ ist oben in zusammengefalteter Form dargestellt („collapsed“)



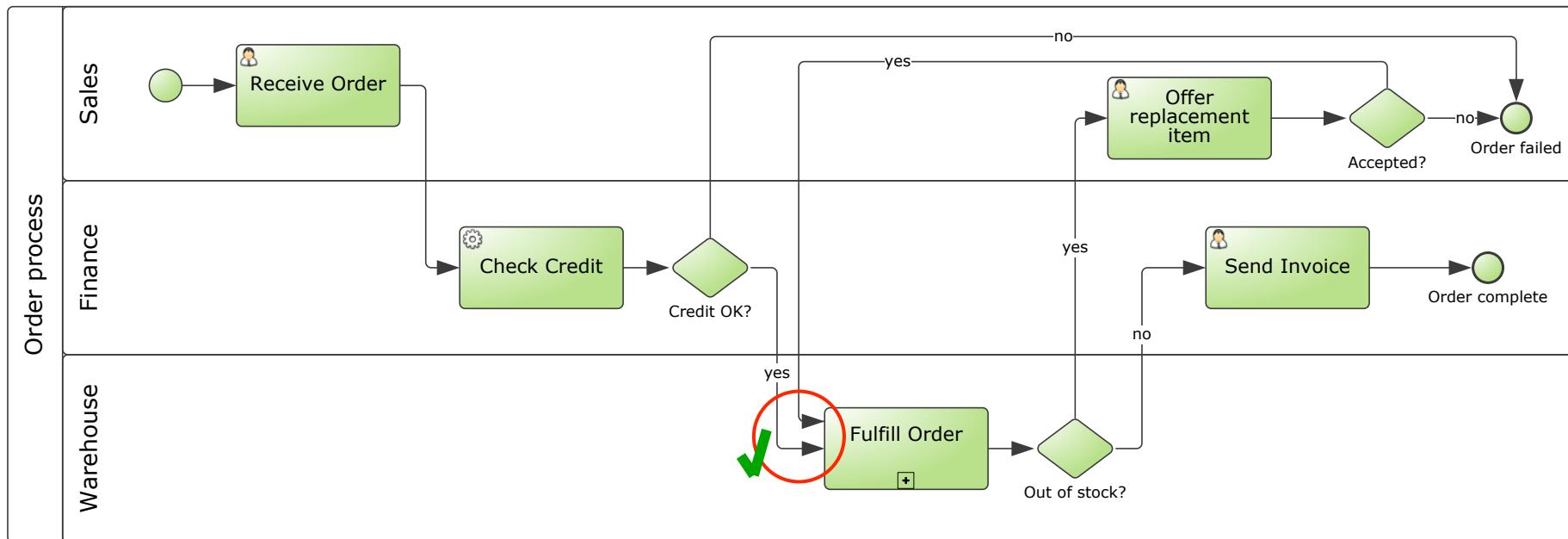
Notation – expanded subprocesses



- der subprocess „Fulfill Order“ in aufgeklappter Form („expanded“)



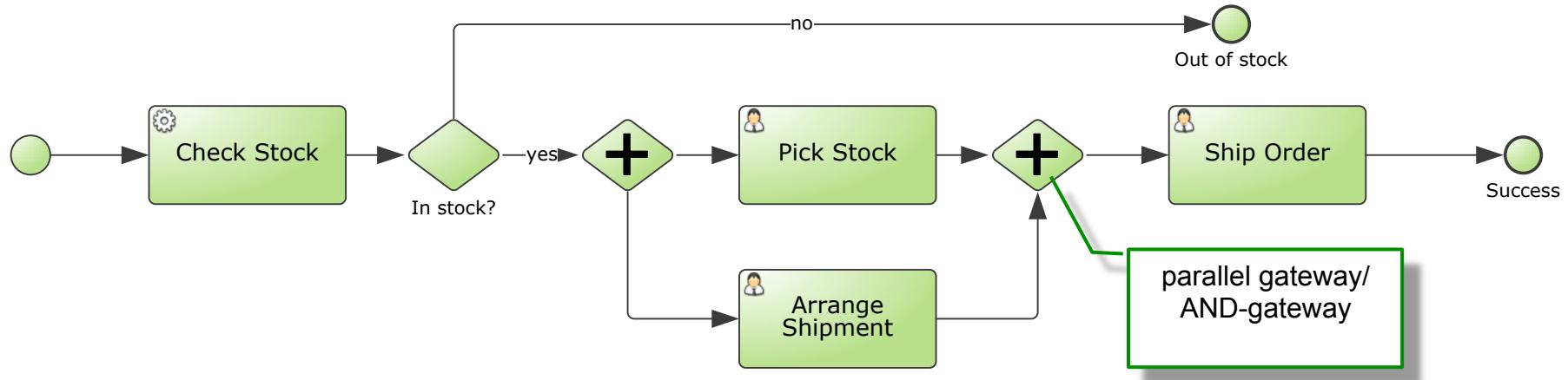
Notation – Schleifen



- Freie Schleifen sind erlaubt („OR-Semantik“ im Ggs. zur AND-Semantik bei UML2-Aktivitätsdiagrammen)



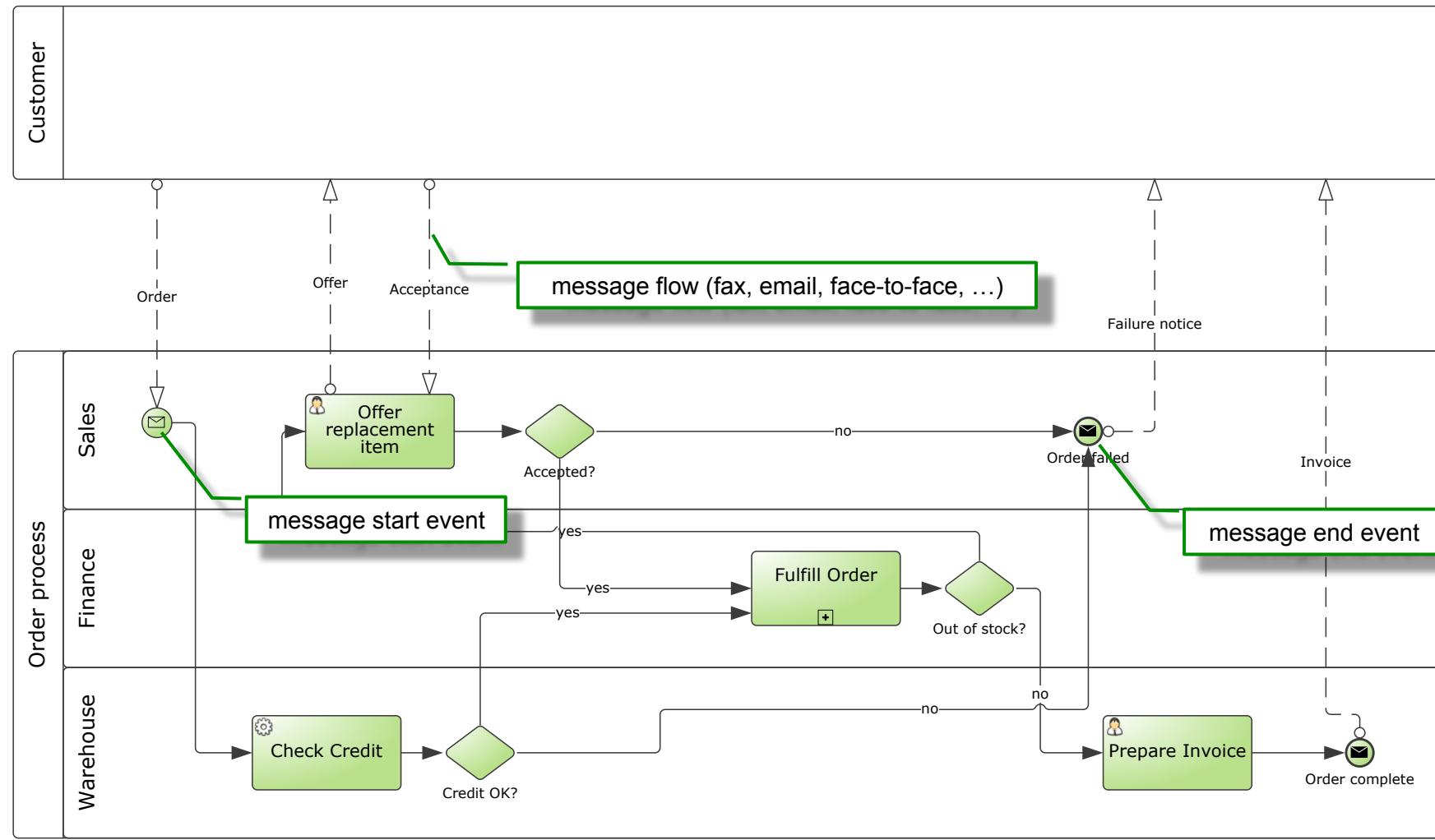
Notation – Parallelität und Synchronisation



- „Pick Stock“ und „Arrange Shipment“ **können** parallel ausgeführt werden, müssen aber nicht
 - falls derselbe Sachbearbeiter sie ausführt, dann ist die Reihenfolge nicht relevant
- Eine spätere Synchronisation ist nicht unbedingt erforderlich
 - Der Prozess terminiert in diesem Fall erst, sobald alle unabhängigen Zweige ein *end event* erreicht haben.
- Vorsicht: nicht verwechseln mit



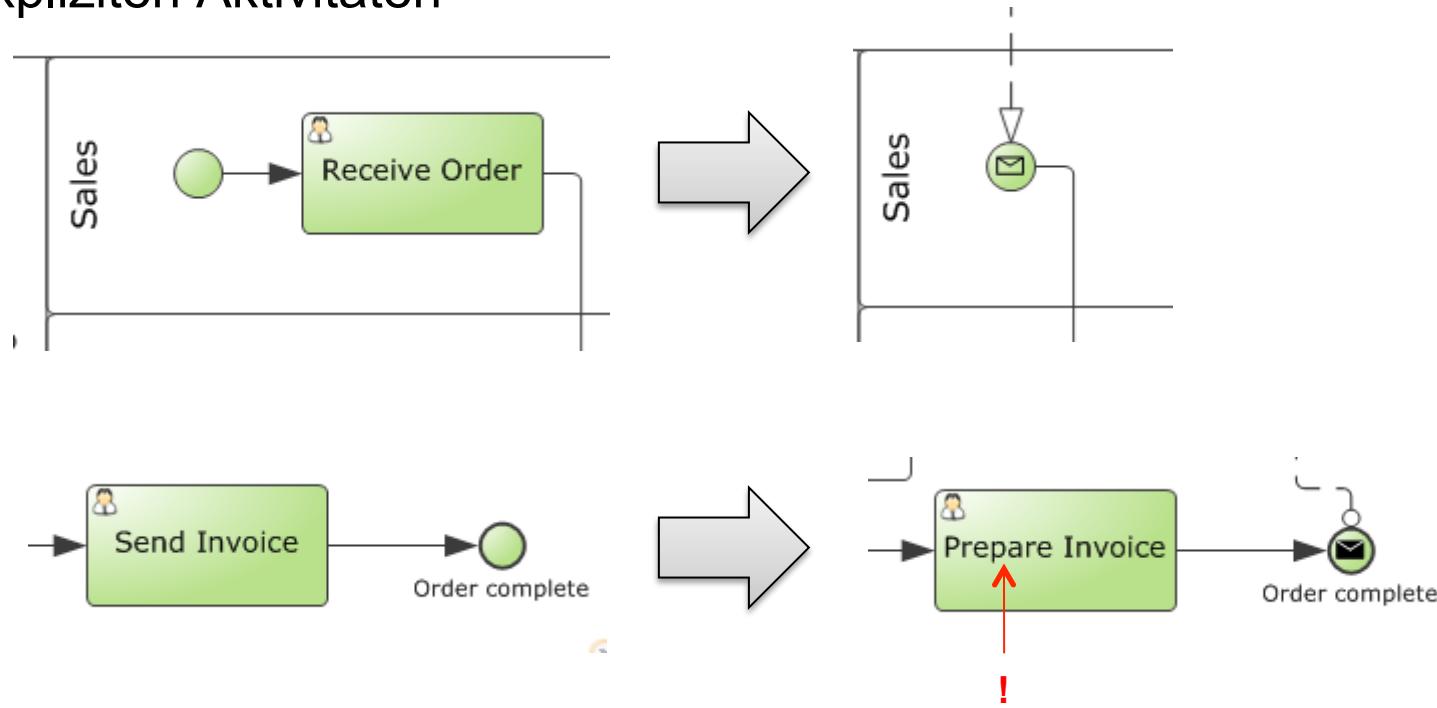
Notation – collaboration diagram (1/3)





Notation – collaboration diagram (2/3)

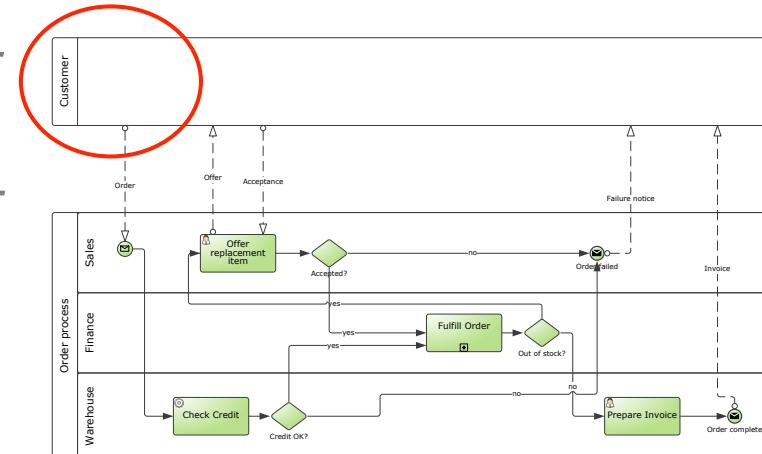
- Verwendung von *message start/end events* im Gegensatz zu expliziten Aktivitäten





Notation – collaboration diagram (3/3)

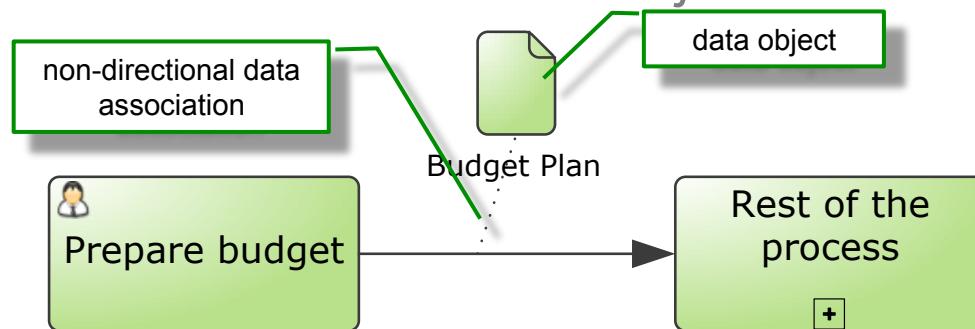
- Der „Kunde“ wie auch andere externe Teilnehmer „interagieren“ mit dem Prozess
- Diese werden mit einem *black-box-pool* modelliert
 - im Ggs. zu unserem *white-box-pool*, welcher unseren Prozess detailliert darstellt
- Die Abläufe dort sind für uns aus Prozesssicht nicht relevant!
- Die Interaktion geschieht über **Nachrichten** (siehe *message start event*) – Mail, Fax, Brief, Anruf, etc.





Notation – Datenfluss (1/2)

- Daten- oder Dokumentflüsse zwischen *activities* und/oder *events* werden mit Hilfe von *data objects* modelliert

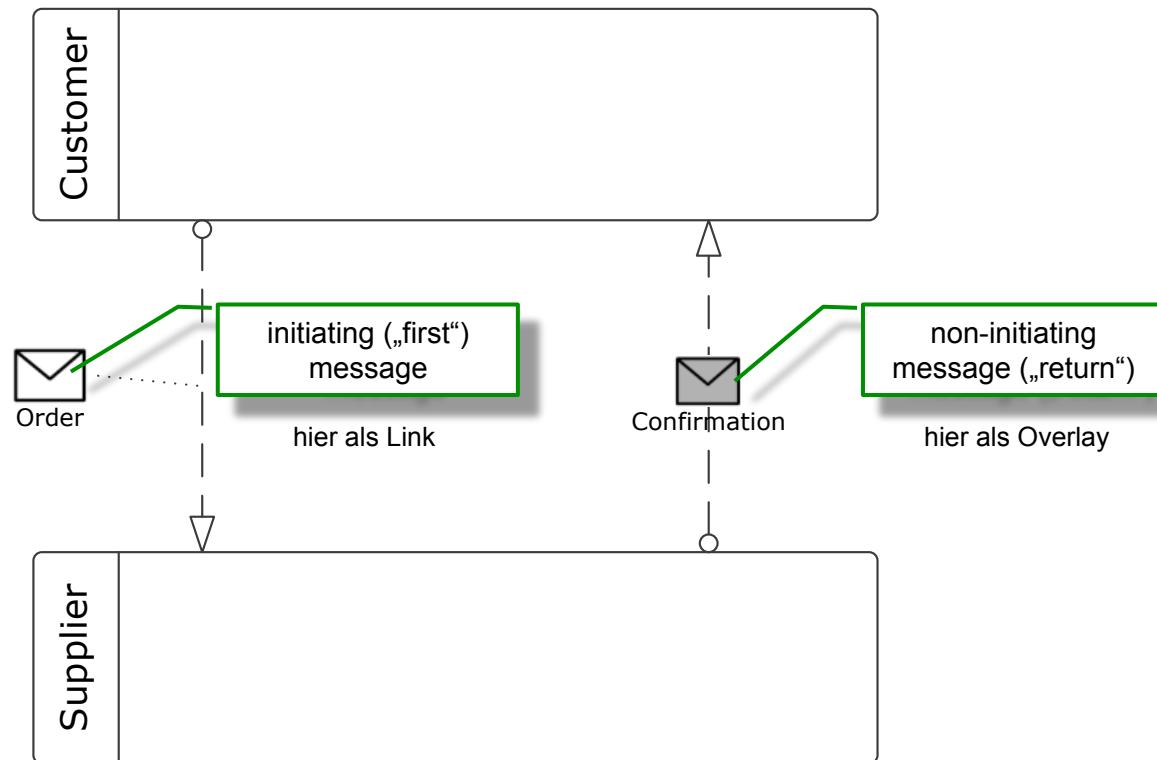


- Optional: Angabe eines Objektzustands
Budget Plan
[accepted]
- data associations* verbinden *data objects* mit Sequenz- oder Nachrichtenflüssen



Notation – Datenfluss (2/2)

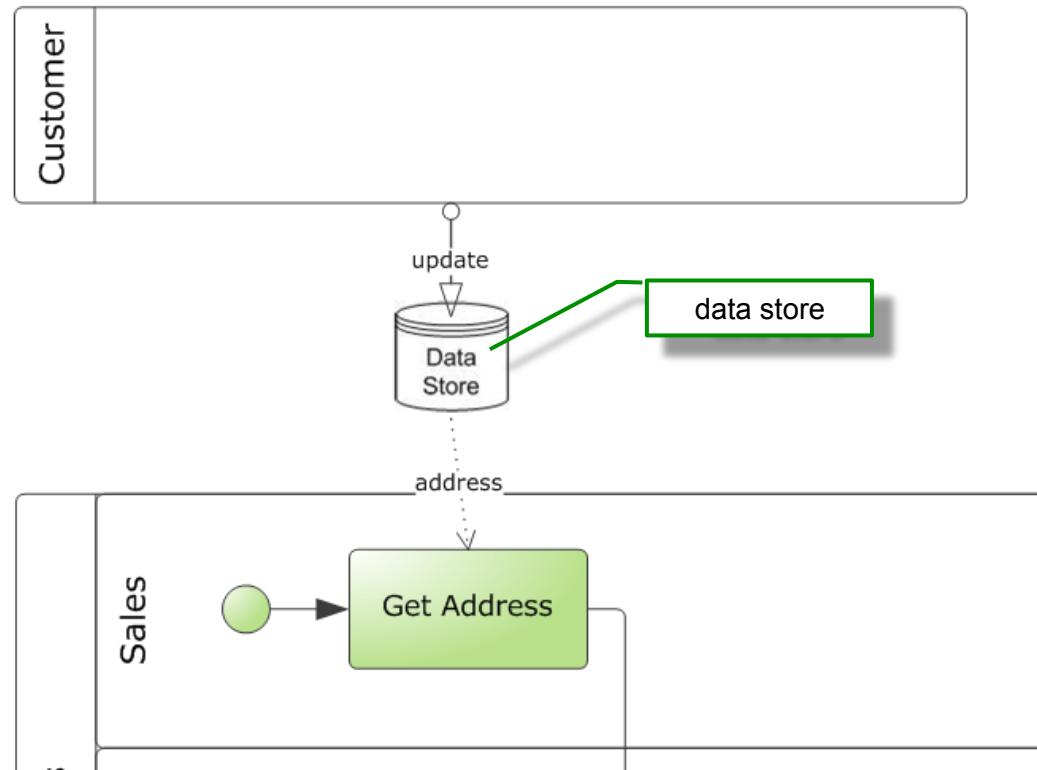
- *data associations an message flows*
 - mit Link oder als Overlay gezeichnet





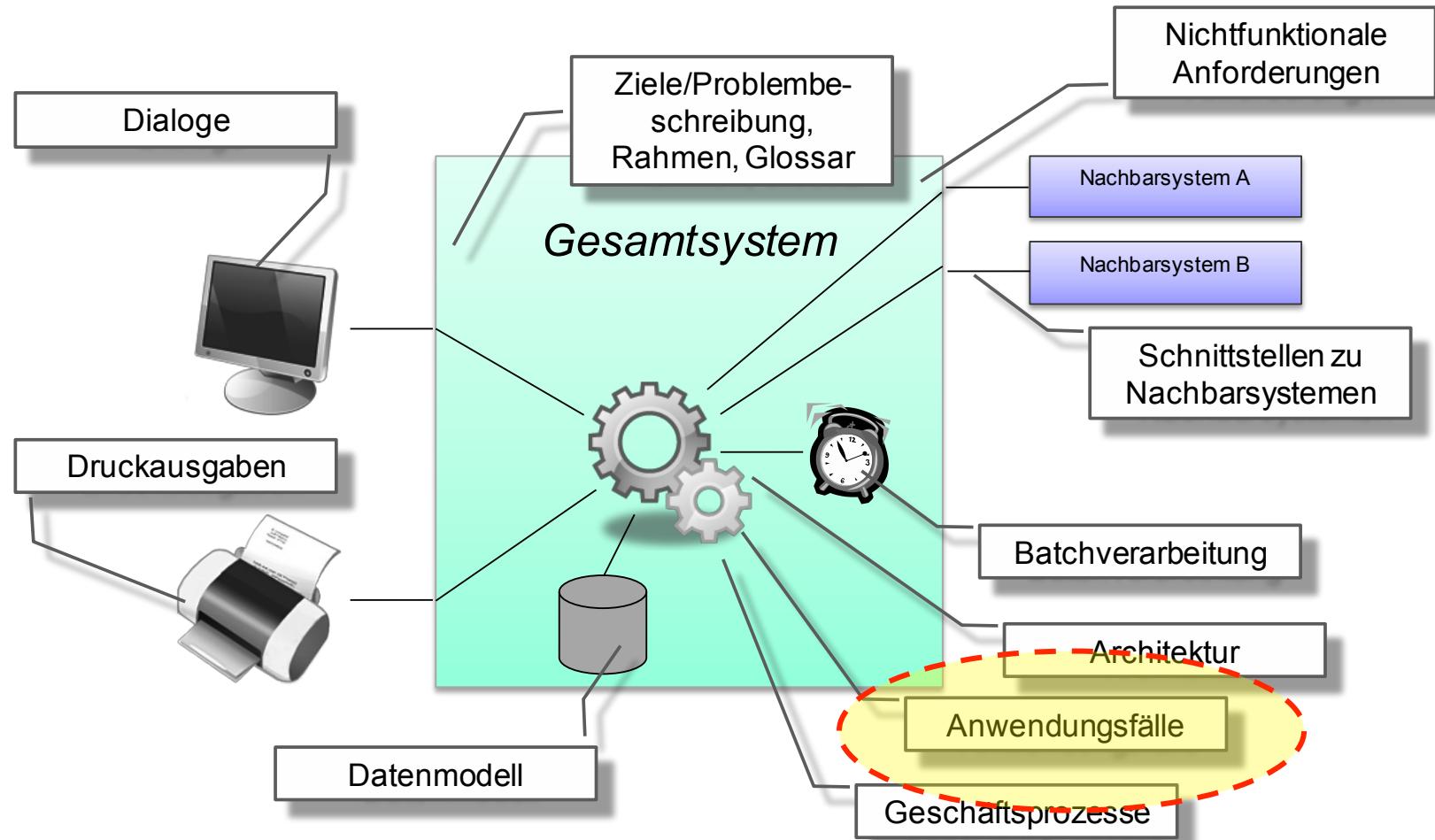
Notation – *data store*

- *data stores* modellieren Dateien, Datenbanken oder (allgemein) Applikationen, mit denen Informationen ausgetauscht werden können



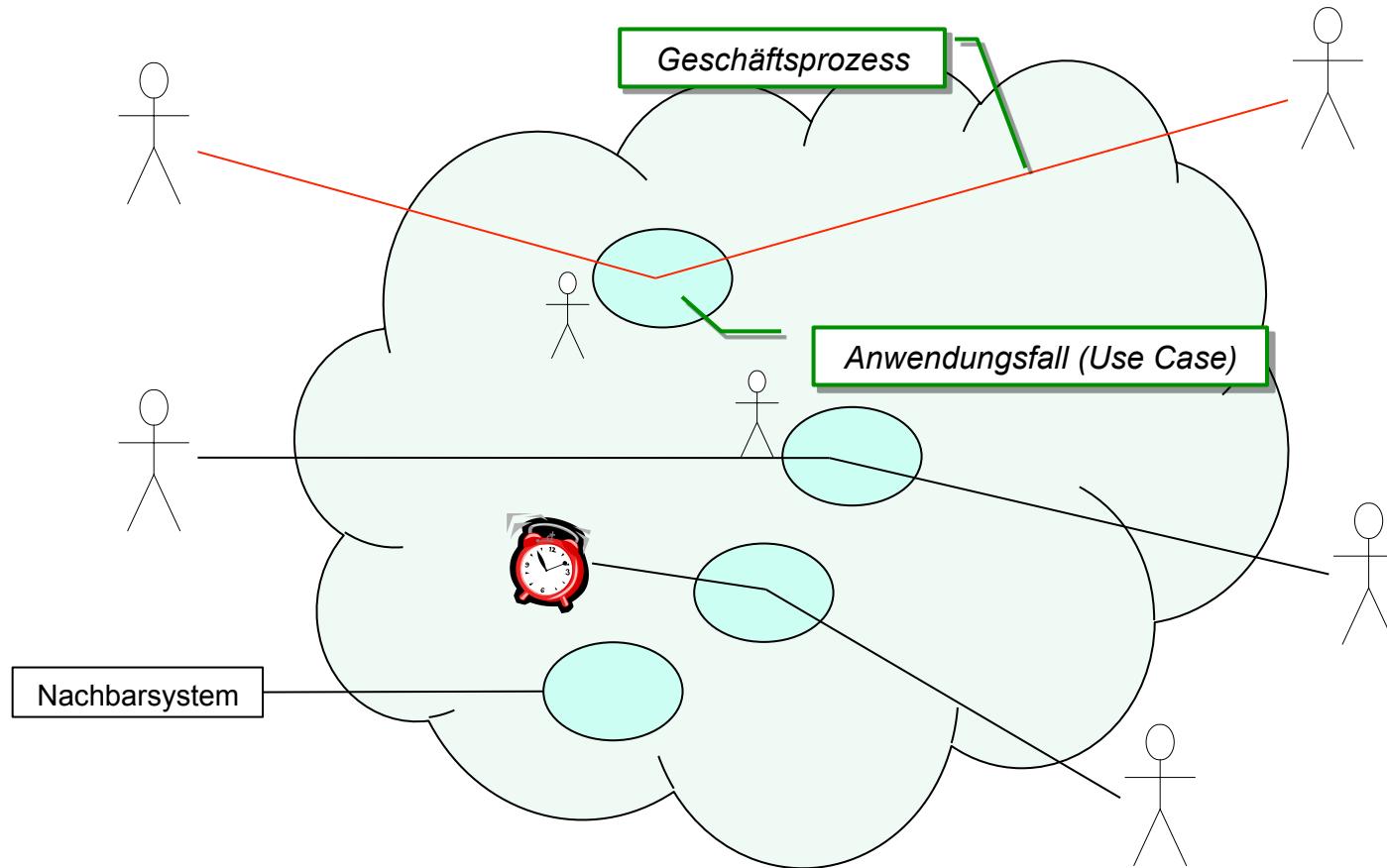


Spezifikation – Anwendungsfälle





Von Geschäftsprozessen zu Anwendungsfällen





Anwendungsfälle

- **Anwendungsfälle** (Use Cases)
 - **Anwendungsfälle** sind diejenigen Arbeitsschritte aus den Geschäftsprozessen, die **systemseitig unterstützt** werden sollen!
 - beschreiben das **nach außen sichtbare Verhalten** des Systems
 - Verhalten und die Interaktion eines Systems als Reaktion auf die zielgerichtete Anfrage oder Aktion eines Akteurs.
 - Wichtig: Jeder Anwendungsfall hat einen direkten Nutzen für den Akteur
 - entstehen aus Geschäftsprozessen und Anforderungen des Kunden
 - Textuelle Beschreibung, evtl. Use Case Diagramm als Übersicht
- **Akteur**
 - Personen, die das System benutzen
 - Rollen, die im System für Benutzergruppen angelegt sind
 - Benachbarte Systeme
- **System**
 - Alles, was wir implementieren



Anwendungsfall – Vorgeschlagene Struktur (1/2)

Titel: Aktien kaufen

Akteur: Kunde

Ziel: Der Kunde kauft Aktien über das Internet

Auslöser:

Ein Kunde entscheidet sich zur Anlage in Aktien.

Vorbedingungen:

Der Kunde besitzt Depot und Freigabe für Aktiengeschäfte, das Anlagekonto ist gedeckt.

Nachbedingungen:

Der Kunde befindet sich in Besitz der Aktien.

Erfolgsszenario:

1. Kunde ruft Bank-Website auf und meldet sich mit Benutzernamen und Passwort an.
2. Kunde navigiert zu Formular für den Aktienkauf.
3. Kunde füllt Formular für Aktienkauf mit den folgenden Daten: WKN, Anzahl.
4. Kunde wählt die Funktion „Kauf“.
5. Das System prüft den Auftrag anhand der Kriterien: Ist die WKN gültig? Liegt die Anzahl zwischen 1 und 100?
6. Das System erstellt Auftragsbestätigung (Inhalte: Datum, WKN, Anzahl, Preis) und zeigt sie dem Kunden an.
7. Das System schickt dem Kunden die Auftragsbestätigung per Email (Inhalte: siehe Punkt 6).

Technikneutrale
Beschreibung des Ablaufs

diese Daten stehen auch im
Datenmodell!!



Anwendungsfall – Vorgeschlagene Struktur (2/2)

Erweiterungen:

1.a Der Kunde besitzt noch kein Online-Profil für sein Depot:

 1.a.1 Ausführung des Anwendungsfalls „Der Kunde erstellt ein Online-Profil“.

...

*Nummerierungsschema beachten. 1a=erste Erweiterung von Punkt 1;
1b=andere Erweiterung von Punkt 1; usw.*

wiederkehrende Funktionalität
in weiteren Anwendungsfällen

Fehlerfälle:

1.a Benutzeranmeldung des Kunden schlägt ein- bis zweimal fehl: System meldet Anmeldungsfehler und fordert zur nochmaligen Eingabe auf.

1.b Benutzeranmeldung des Kunden schlägt dreimal in Folge fehl: System sperrt Benutzerkennung vorläufig und sendet E-Mail-Benachrichtigung an Kunde.

5.a Falls WKN ungültig: System stellt Formular dar (mit vorher eingegebenen Werten), markiert den Fehler („Die angegebene WKN existiert nicht“) und fordert den Kunden zur Korrektur auf.

5.b Falls Anzahl nicht zwischen 1 und 100: ...

...

Häufigkeit:

Pro Kunde zwischen einmal jährlich bis zehnmal monatlich bei einer durchschnittlichen Kundenanzahl von 100.000 Kunden.

Anforderungen:

A10, A11, A20, A32



Anwendungsfälle – Granularität

- Vorsicht mit der Granularität:
 - häufige Fehler:
 - zu grobgranulare Use-Cases (zu komplexe oder langandauernde Interaktion)
 - zu feingranulare Use-Cases (einzelner Schritt oder Unterfunktion in einem Geschäftsablauf)
 - Beispiele:
 - **gut**: „Bestellung aufnehmen“, „Buch ausleihen“, „Karten reservieren“, ...
 - **schlecht**: „einen Vertrag aushandeln“ (dauert lange und involviert viele Personen), „Benutzerprofil anzeigen lassen“, „Dokument ausdrucken“ (beide zu feingranular)



Aufgabe: Anwendungsfall (Use Case)

Anwendungsfall

Formulieren Sie einen möglichen Anwendungsfall „Kinokarten im Internet reservieren“ (inkl. Auswahl der Vorstellung, Bezahlung, etc.)



15 Minuten



Anwendungsfalldiagramme

Anwendungsfalldiagramme stellen die **Anwendungsfälle** des Systems zusammen mit den beteiligten **Akteuren** in grafischer Form dar.

Sinnvoll als

- Einstiegspunkt für die Spezifikation.
- Kommunikation auch mit Kunden.

Nicht sinnvoll

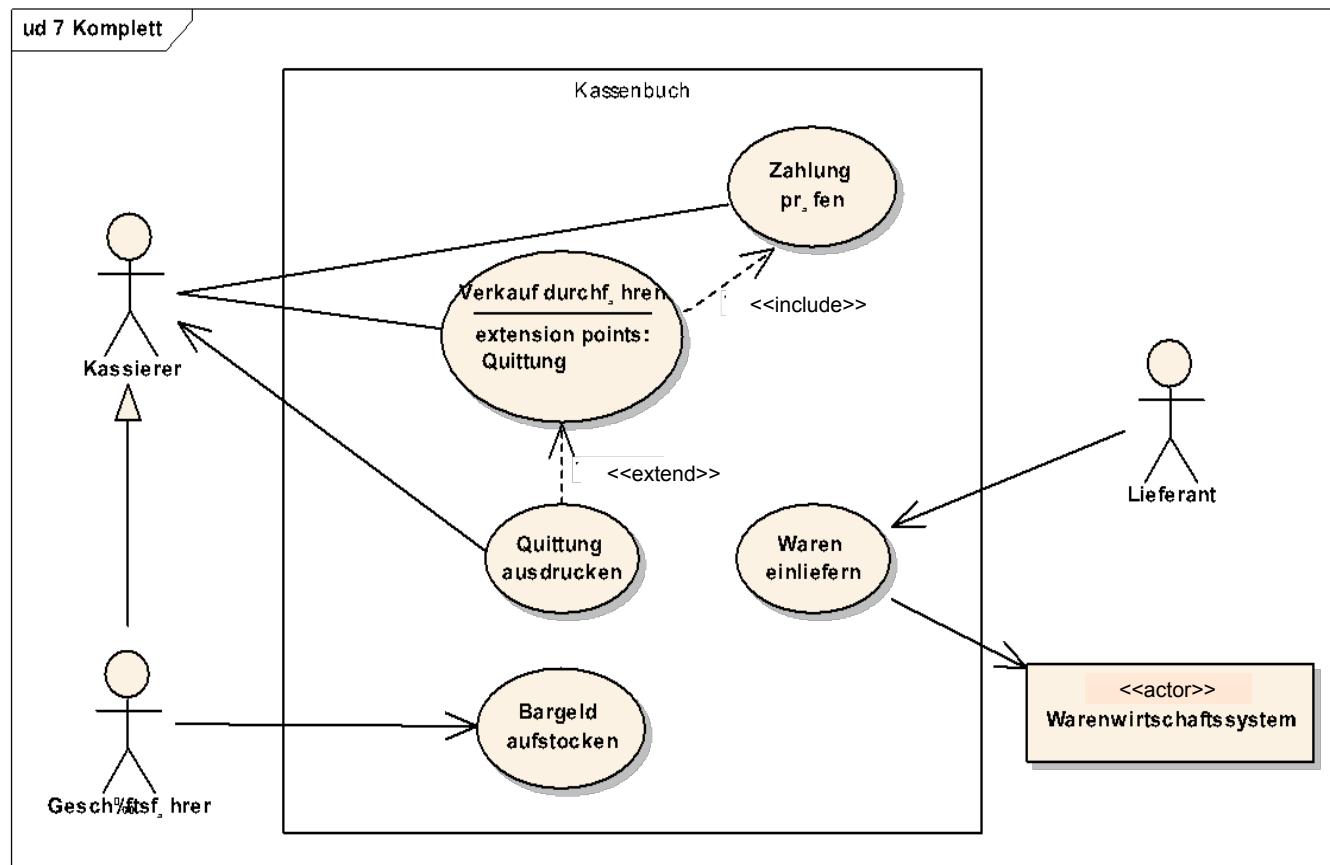
- zum Zeigen *aller* Anwendungsfälle!

Und: Ein Diagramm alleine macht noch keinen Anwendungsfall!



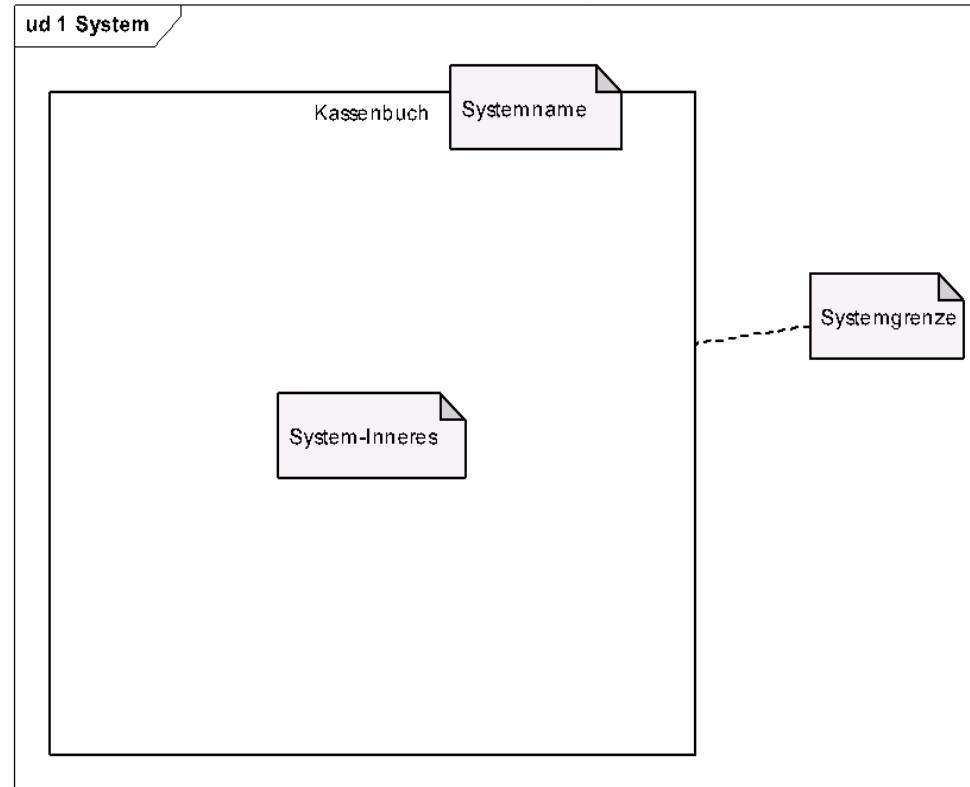


Beispiel





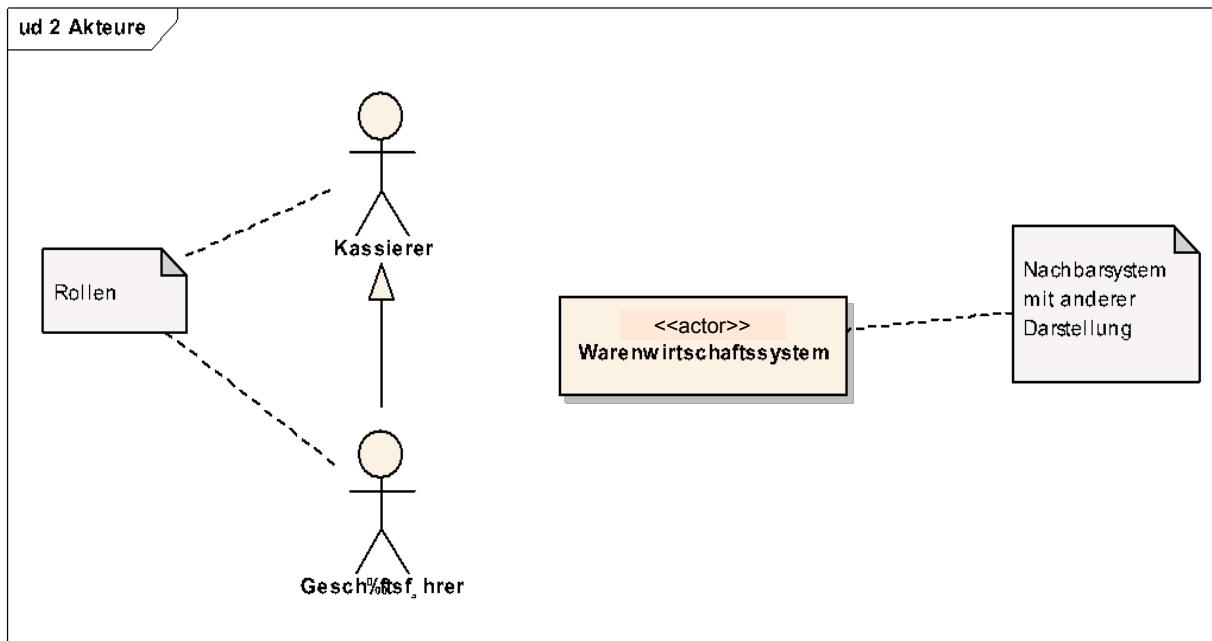
Notation: System



- Umschließt das gesamte betrachtete System
- Aber nicht die gesamte Software oder Nachbarsysteme



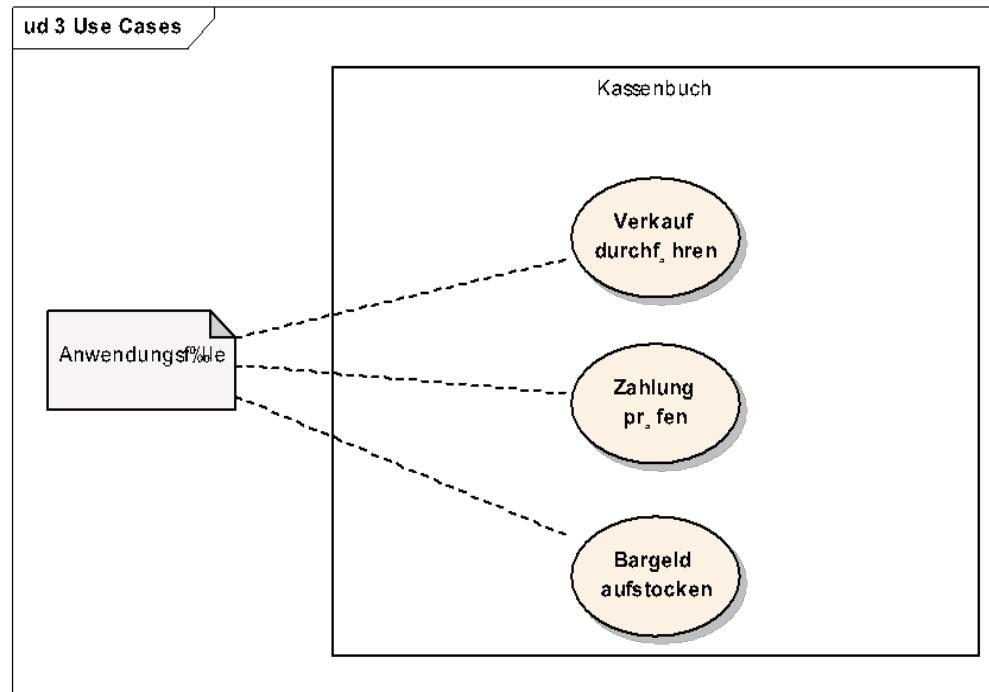
Notation: Akteure



- Außerhalb des Systems, meist links und/oder rechts
- Rollen, Nachbarsysteme
- Keine Beziehung zum System



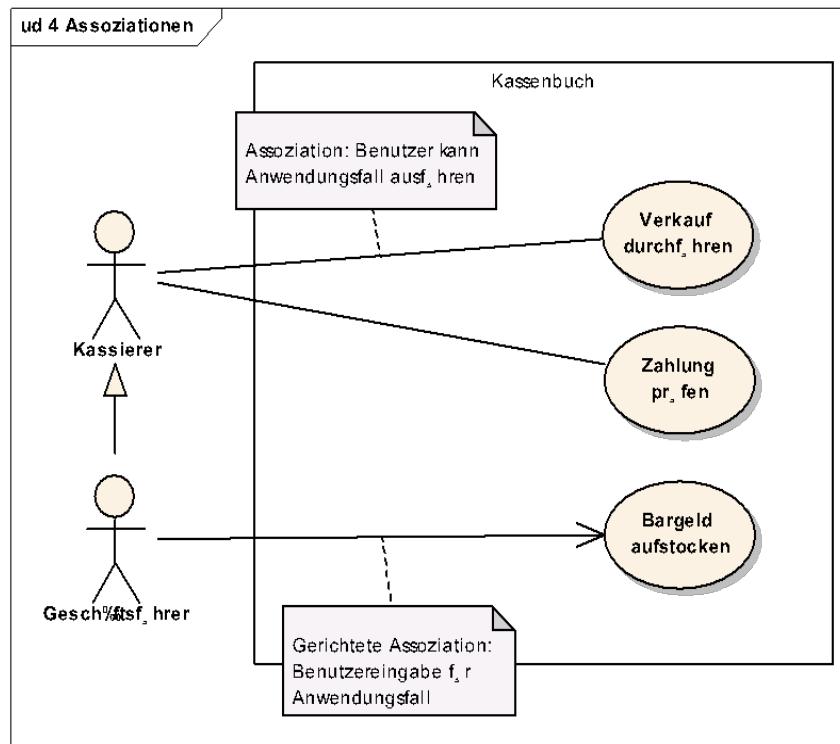
Notation: Anwendungsfälle



- Alle innerhalb des Systems, schließlich werden sie vom System realisiert
- Namensgebung für Anwendungsfälle: **Objekt + Verb im Infinitiv**



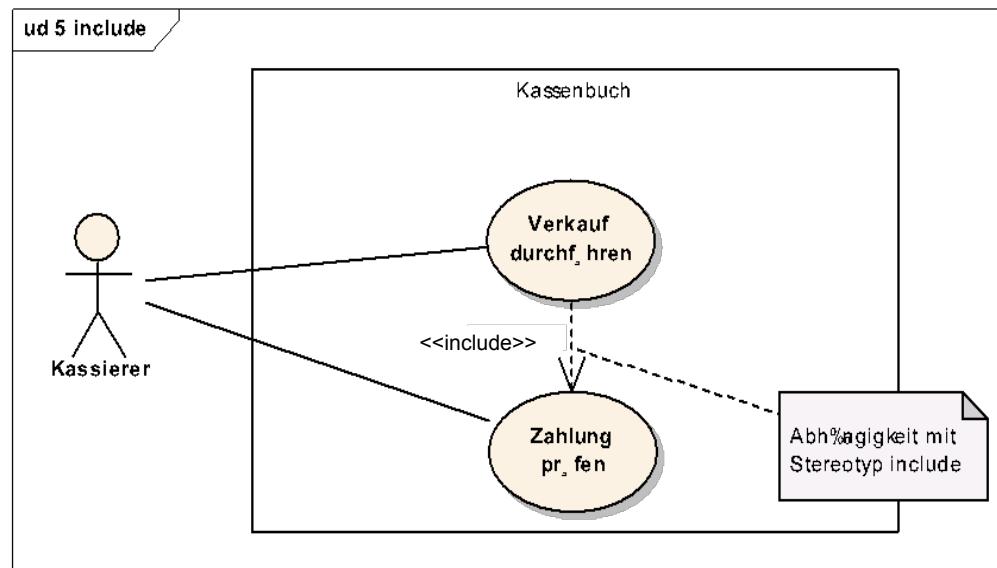
Notation: Assoziationen



- Wenn ein Anwender einen Use Case durchführen kann/darf, wird eine einfache Assoziation zwischen beiden gezeichnet
- Für Ein- und Ausgabeflüsse gerichtete Assoziationen
- Rollenbeziehungen können angegeben werden; ein Geschäftsführer „ist ein“ Kassierer



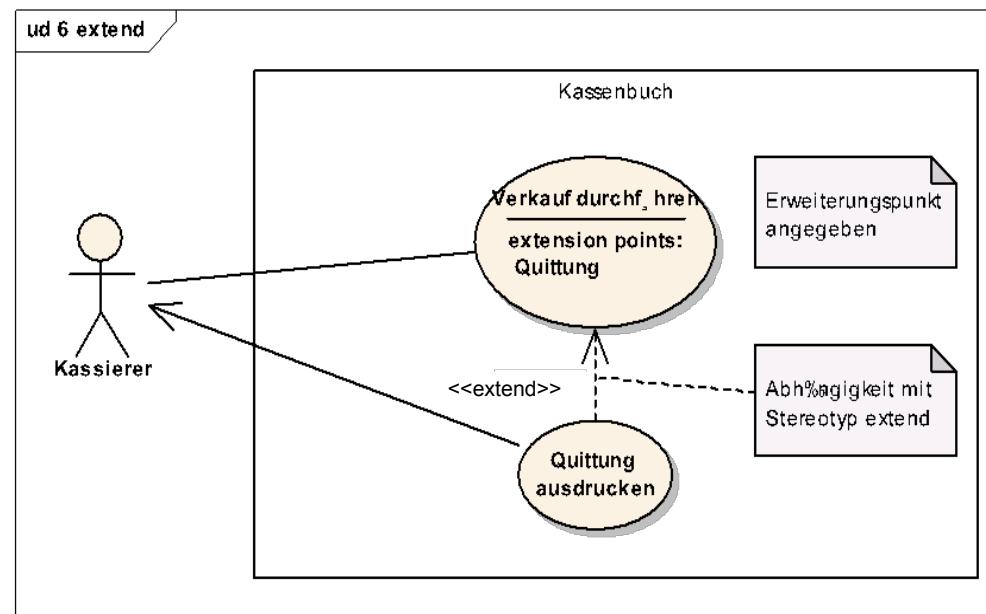
Notation: «include»



- Anwendungsfall A beinhaltet immer Anwendungsfall B und wird während der Abarbeitung von A ausgeführt
- Weiteres Beispiel: Während einer Buchung muss immer ein Beleg eingegeben werden.



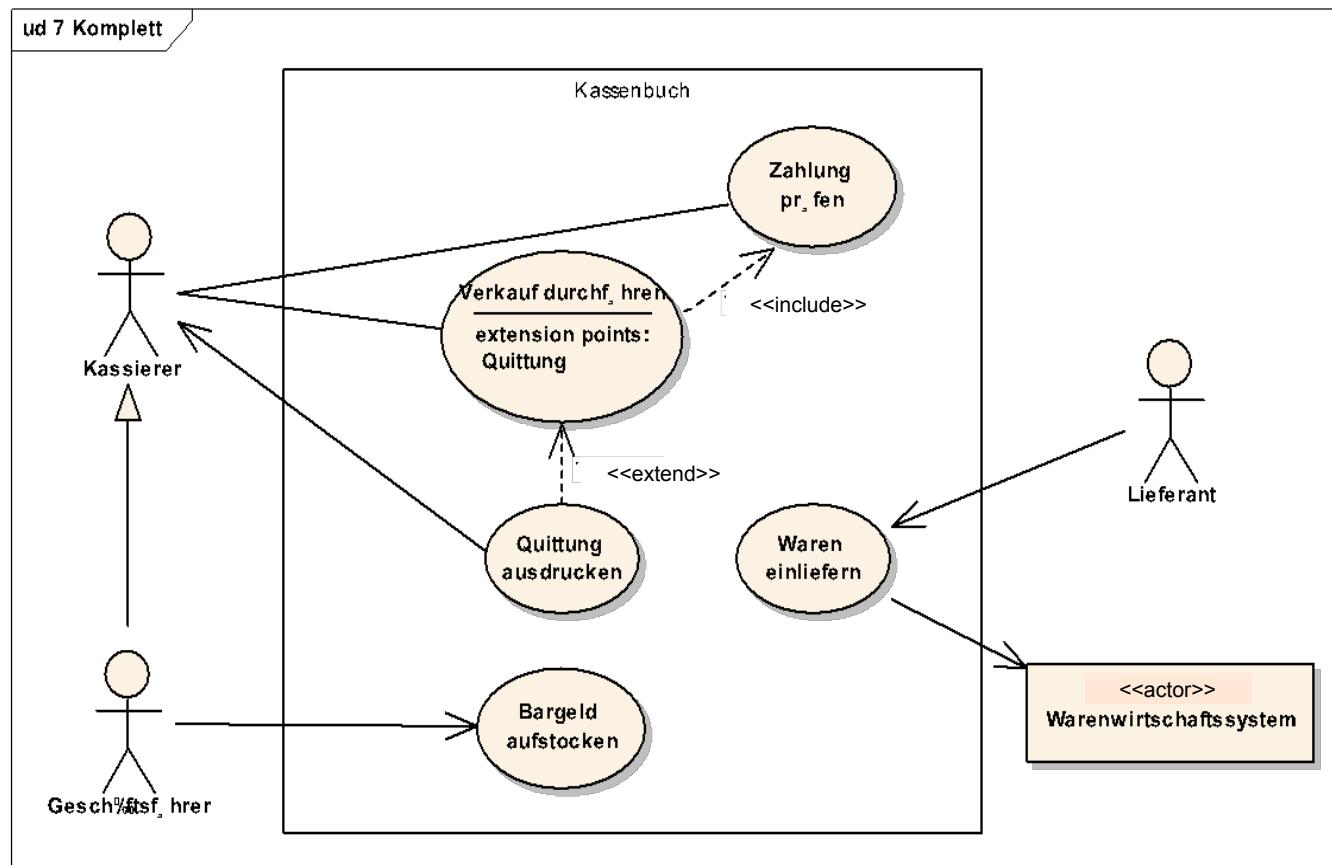
Notation: «extend»



- Anwendungsfall B ist optionaler Bestandteil von Anwendungsfall A
- Extension Point immer notwendig
- Weiteres Beispiel: Bei einer Buchung kann ein Steuersatz angegeben werden



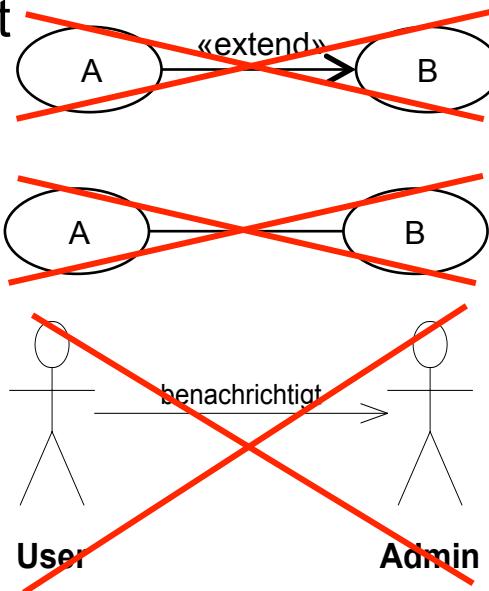
Beispiel revisited





Typische Fehler

- Zu feingranulare Anwendungsfälle
- Diagramme ohne begleitenden Text sind wenig aussagekräftig!
Ohne textuelle Beschreibung ist dies sinnlos.
- Use Case Diagramme sind nicht „objektorientiert“!
- Diagramme mit Use-Cases und Beziehungen überfrachtet
- <> ohne Erweiterungspunkt
- Assoziationen zwischen Use-Cases
- Assoziationen zwischen Akteuren
- Use Cases außerhalb der Systemgrenze





User Stories / Epics

- **User Stories** werden oftmals in agilen Vorgehensweisen verwendet (→ später) und stellen in etwa „leichtgewichtige“ Anwendungsfälle dar.
- Eine User Story („Benutzergeschichte“) ist eine in Alltagssprache formulierte Software-Anforderung.
- Sie ist bewusst kurz gehalten und umfasst in der Regel nicht mehr als zwei Sätze.
- Beispiele:
 - „*Ich, als Student, kann mein monatliches Parkticket online kaufen.*“
 - „*Ich, als Professor, kann Noten online eintragen und verändern.*“
- Als erste grobe Anforderungsbeschreibung nützlich, allerdings im Weiteren Verlauf des Projekts zu detaillieren (z.B. durch Anwendungsfälle).
- **Epics** sind grobgranulare User Stories, die zunächst in User Stories zerlegt werden müssen.



Use Cases oder User Stories?

- Alle Systemfunktionen mit Hilfe von Use Cases (Anwendungsfällen) zu spezifizieren kostet zu viel Aufwand.
- **Sinnvoll:**
 - Hauptfunktionalitäten wie bspw. „*Kinokarten reservieren*“ (geschäftskritisch, umfangreich, wichtig für den Kunden und/oder für uns, ...) mit **Use Cases** formulieren
 - Nebenfunktionen („*Benutzerprofil anzeigen lassen*“, „*Login*“) mit **User Stories**