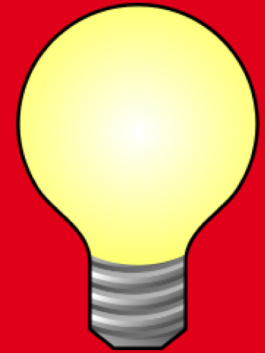


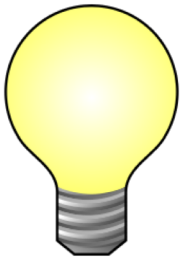


Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

26.11.2020
Zustandsdiagramme

Zustandsdiagramme





AGENDA



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

Einführung ins Thema

Loslegen mit einem Beispiel

Modellelemente im Überblick

Weitere Anmerkungen

Fazit



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

01 EINFÜHRUNG INS THEMA

Ziel:
Die Eckpunkte des Themas kennenlernen



ZUSTANDSDIAGRAMME



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

- Eignen sich zur Modellierung von Systemen/SW, die:
 - in Zuständen verharren
 - und diese meist durch ein Signal von außen wechseln
- Zustand: Unter Umständen verbleibt ein System sehr lange in einem bestimmten “Modus” → Zustand
- Wechsel: Meist durch ein Signal von außen
 - Es kann sehr lange dauern bis ein Wechsel eintritt

ZUSTANDSDIAGRAMME



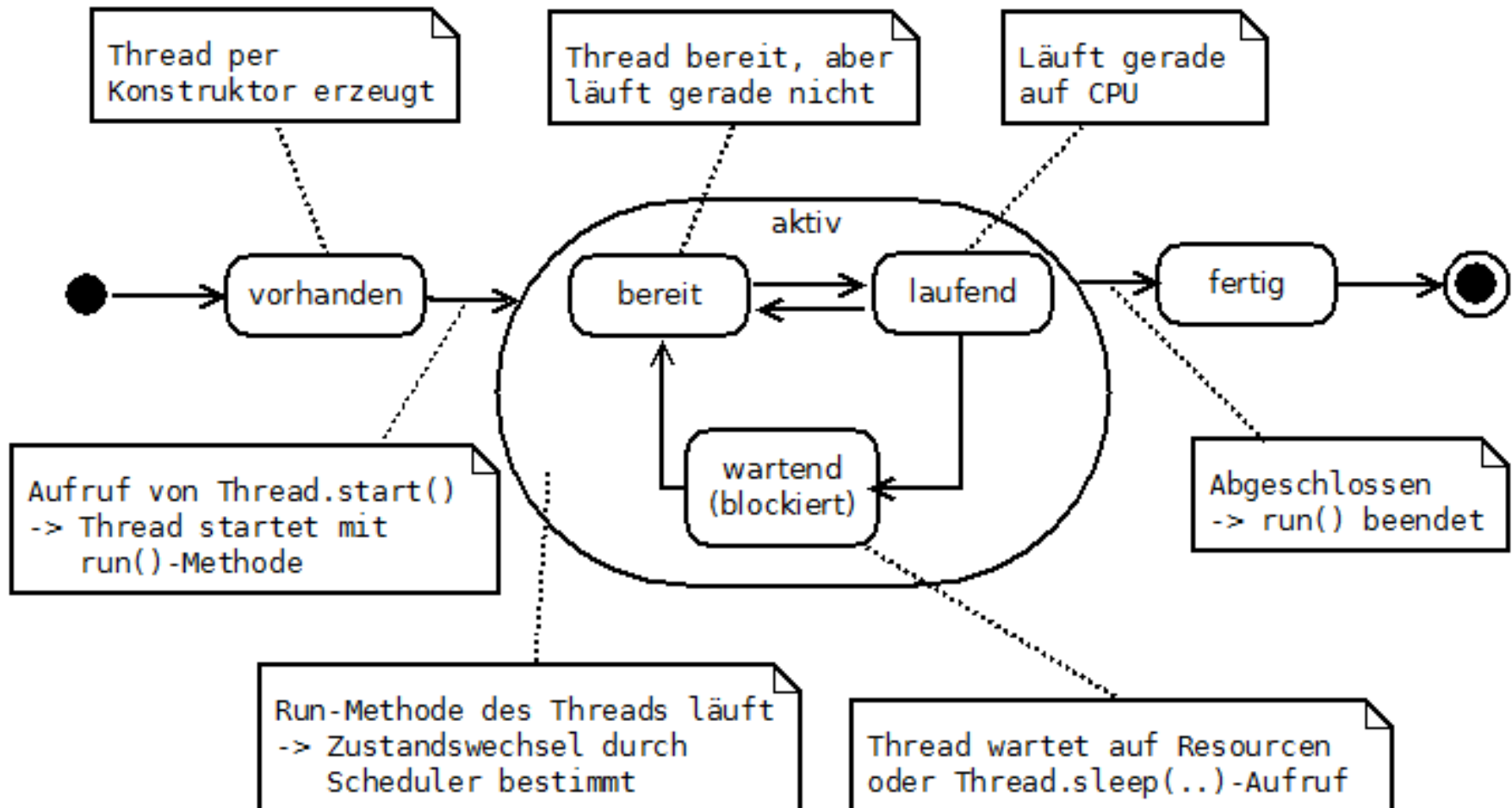
Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

- Zustand: Unter Umständen verbleibt ein System sehr lange in einem bestimmten “Modus” → Zustand
- Wechsel: Meist durch ein Signal von außen
 - Es kann sehr lange dauern bis ein Wechsel eintritt
- Beispiele:
 - Getränkeautomat:
 - Zustände: Wartend, Geld eingeworfen, Getränkewahl, Bereitstellung Getränk, Wartend
 - Automobilsteuergeräte:
 - Zustände: Schlafen (23h am Tag) –(Tür öffnet sich)→ aufgewacht
Zündschlüssel steckt, Motor gestartet, Motor aus,
Zündschlüssel raus –(Tür zu)→ Schlafen

2. SEMESTER - PM: MÖGL. ZUSTÄNDE EINES THREADS



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim



ANDERE NAMEN FÜR ZUSTANDSDIAGRAMME



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

- State Machine, Zustandsautomat (endlich)
- State Diagram
- State Chart
- Mealy-Automat (Aktion im Übergang)
- Moore-Automat (Aktion im Zustand)

→ Details lernen Sie hier noch in der Vorlesung
“Automatentheorie und Formale Sprachen”



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

02

Loslegen mit einem Beispiel

Ziel:

Erste Sachen kennenlernen

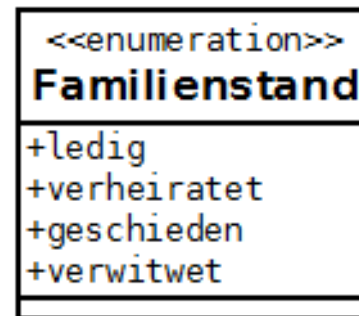
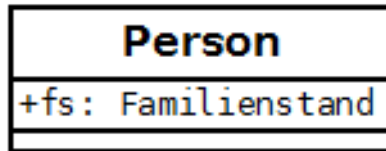


BSP: PERSONEN AUS SICHT DES STANDESAMTS



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

Klassendiagramm:

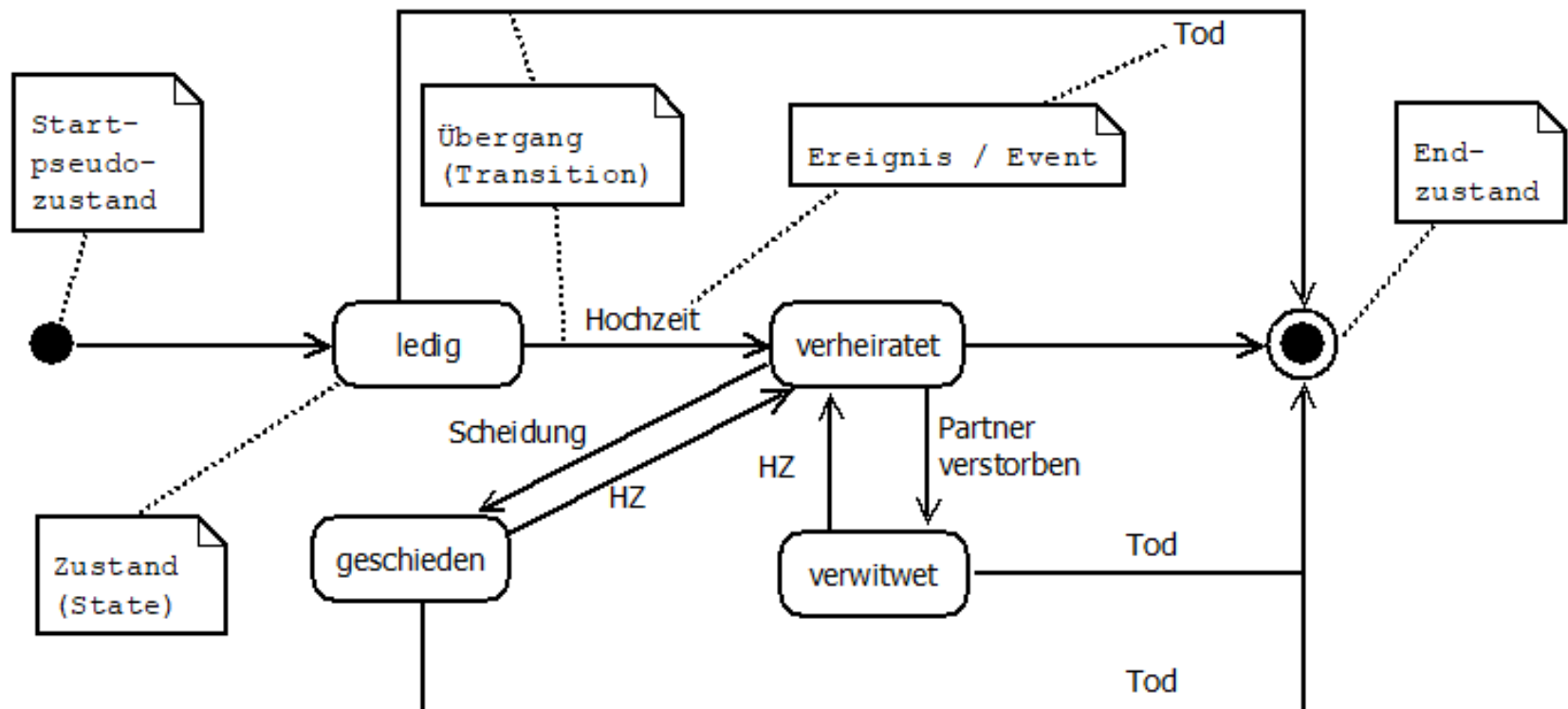
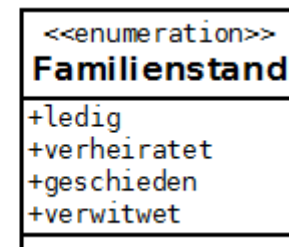
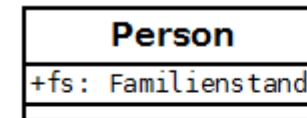


BSP: PERSONEN AUS SICHT DES STANDESAMTS



Hochschule RheinMain
University of Applied Sciences
Wiesbaden Rüsselsheim

Zustandsdiagramm:

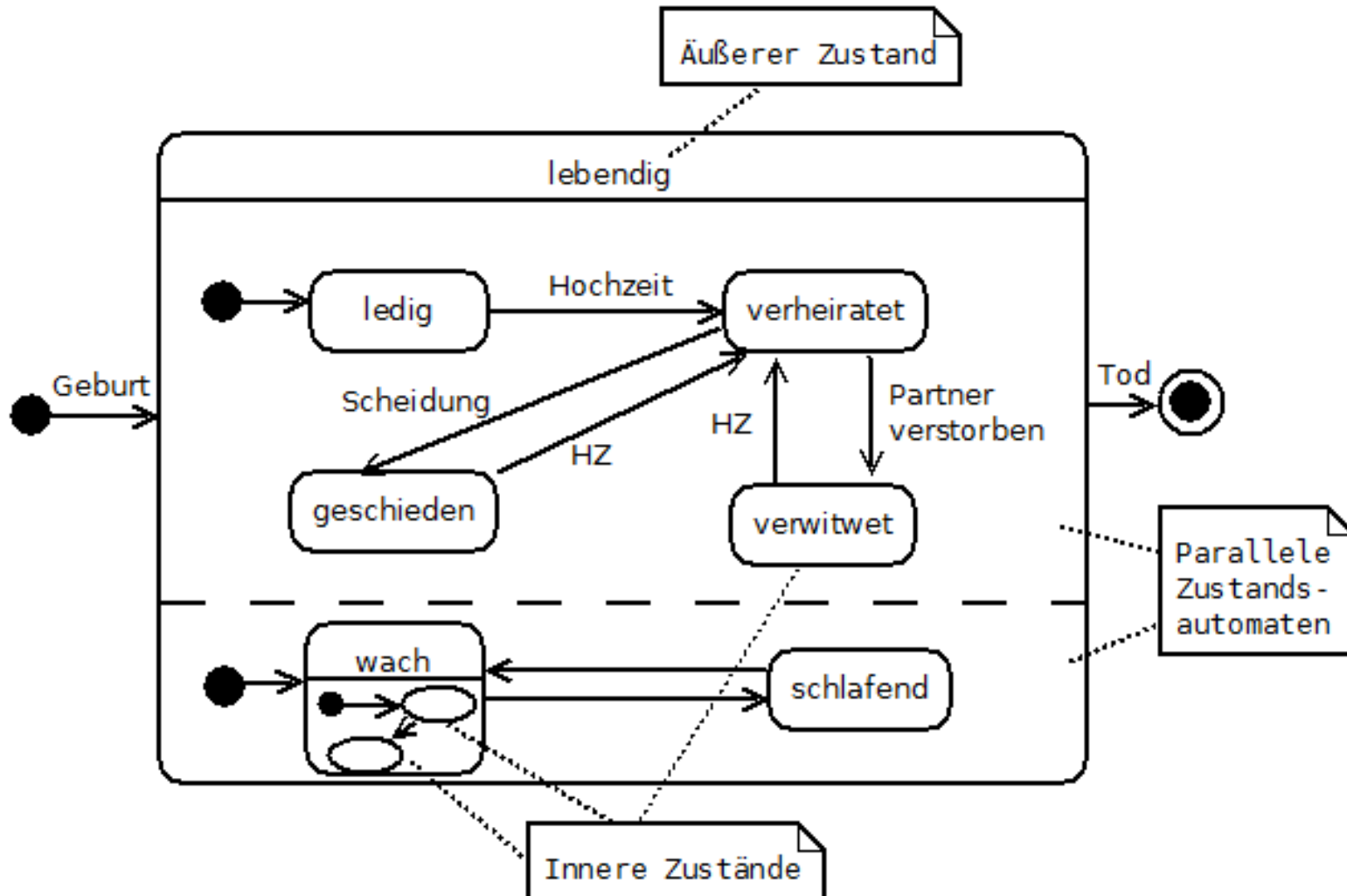


BSP: PERSONEN AUS SICHT DES STANDESAMTS



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

Zustandsdiagramm – Nebenläufigkeit / Schachtelung :





Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

03

Zustandsdiagramme

- Modellelemente im Überblick

Ziel:
Die Elemente im Überblick erfassen



MODELLELEMENTE – ZUSTÄNDE



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

- “Normaler Zustand” (State):



- Pseudozustände (Pseudostate):

- Start



- Einfache Historie



(merkt sich eine Ebene)

- Tiefe Historie



(merkt sich alle Zustände über gesamte Schachtelungstiefe)

- End-Zustand (FinalState)



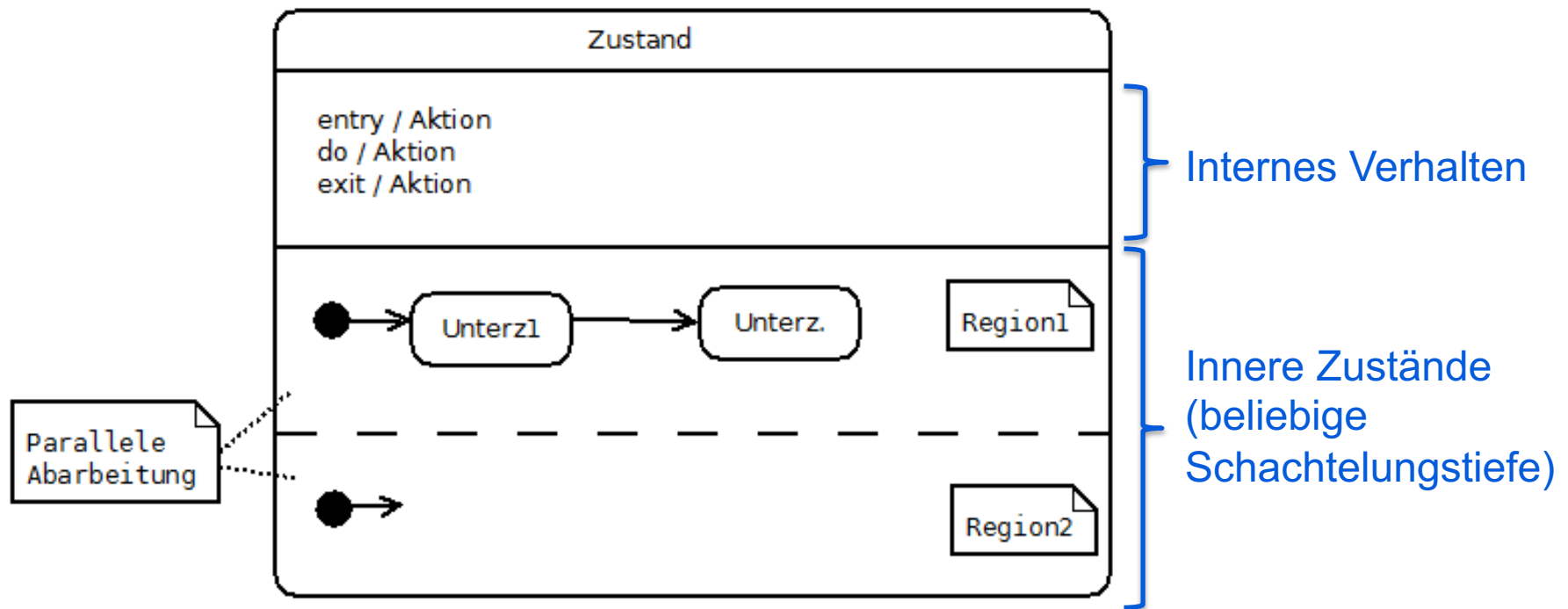
→ Sonderfall, weil er beendet

MODELLELEMENTE – ZUSTÄNDE



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

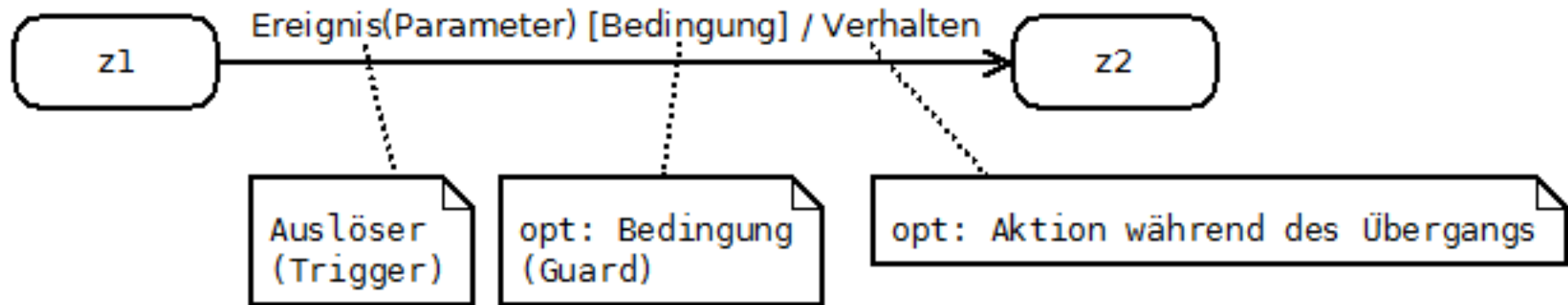
- Möglicher innerer Aufbau von Zuständen:



MODELLELEMENTE – ÜBERGÄNGE



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim



Mögliche Impl. in Java:

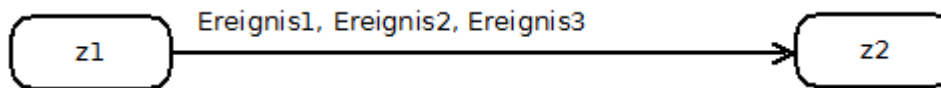
- Trigger \triangleq Methoden-Signatur
- Verhalten \triangleq Methoden-Körper + Guard

MODELLELEMENTE – ÜBERGÄNGE

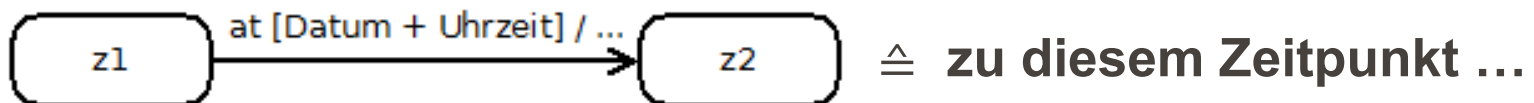
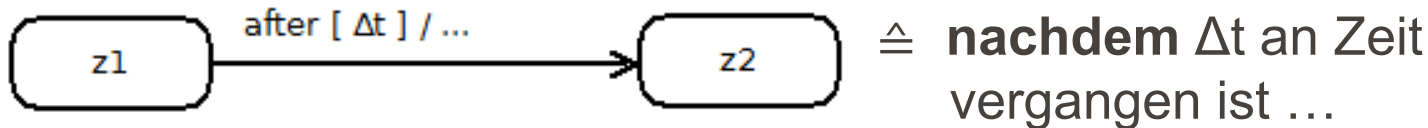


Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

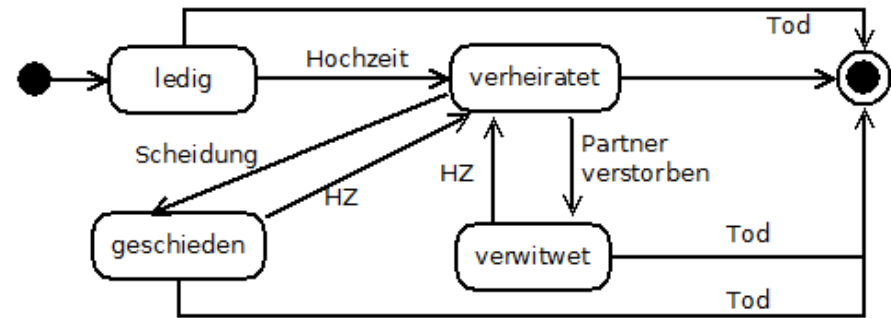
- Mehrere Ereignisse für einen Übergang:



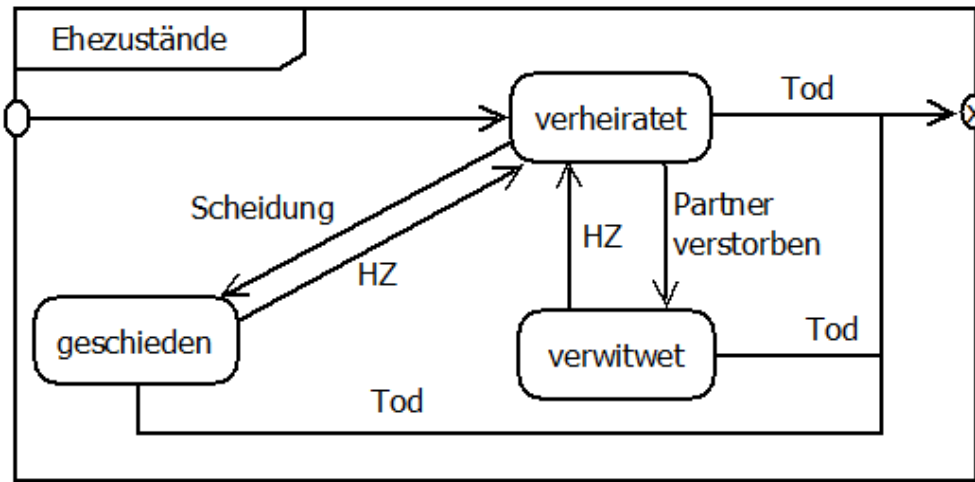
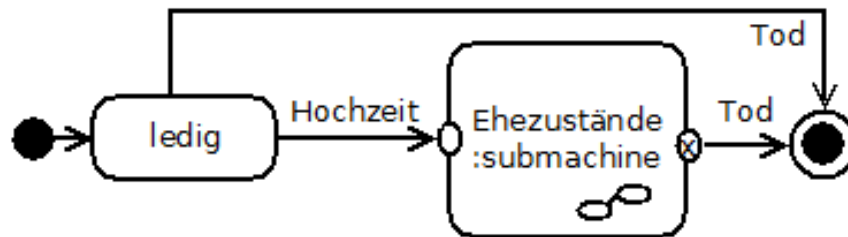
- Spezielle Auslöser (Trigger):



ZERLEGUNG IN UNTERDIAGRAMME



Zustandsdiagramme können auch in mehrere zerlegt werden:



TIPPS ZUR MODELLIERUNG



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

- Name eines Zustands == Adjektiv
(oder Substantiv+Adjekt – z.B.: LichtAn, LichtAus)
- Erst grob, dann Details
 - Zustände sammeln
 - Nach auslösenden Ereignissen suchen
 - Iterativ verfeinern
- Erst auf einer Ebene arbeiten
→ Später ggfs. Schachteln



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

04

Weitere Anmerkungen

Ziel:
Weitere Sachen



WOFÜR EIGNEN SICH ZUSTANDSAUTOMATEN?

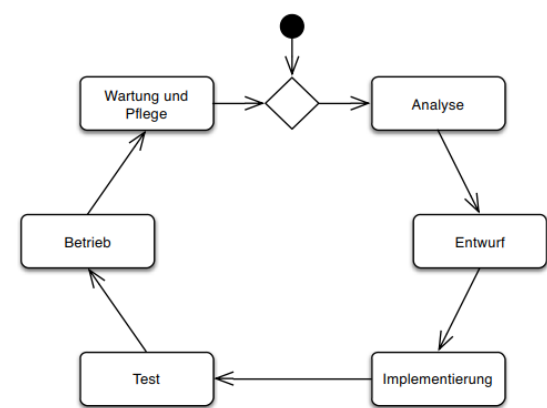
■ Analyse

- Anforderungen des Kunden analysieren
 - Potentielle Zustände des Zielsystems
 - Anderes noch unbestimmtes
- Prozesse oder Systeme analysieren in die das Zielsystem eingebettet ist

■ Design

- Zustände des Zielsystems
- Zustände einzelner Objekte
- Lebenszyklus von Objekten
- ...

■ Implementierung → siehe Design



BEVORZUGTE EINSATZGEBIETE



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

- Asynchrone Vorgänge:
 - Auf eine kurze Aktion folgt eine lange Wartezeit
- Lebenszyklen über mehrere Use Cases hinweg
- Ereignisgesteuerte Vorgänge: z.B. GUI (1 Klick \triangleq 1 Ereignis)
- Verhalten hängt von Vorgeschichte/aktuellem Zustand ab
 - Z.B. Einfügen nicht möglich, wenn Liste schon voll
- Codegenerierung* von zustandbasiertem Verhalten
 - Besonders geeignet für ereignisbasierte Systeme, da Zustandsdiagramme eine deterministische und vollständige Modellierung erlauben
 - Erlaubt dann eine ausgiebigere Analyse mit Simulation, ...

Siehe später
Analysephase

* Siehe z.B.: <https://www.embedded-software-engineering.de/codegenerierung-was-man-damit-nicht-machen-kann-a-555705/>

ZUSTANDSAUTOMATEN ALS TABELLEN



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

- Die in Zustandsautomaten enthaltene Information kann auch in Tabellenform spezifiziert werden (kein UML!)

Ausgangszu stand	Ereignis	Bedingung	Aktion	Zielzustand
				ledig
ledig	Hochzeit			verheiratet
...				

WIE ZUSTANDSAUTOMATEN IMPLEMENTIEREN?



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

Verschiedene Ansätze zur Implementierung:

- Zustandsattribut (z.B. Enum) – Zwei mögliche Implement.:
 1. 1 Trigger \triangleq 1 Methode
 2. Switch – Case
- Tabelle
 - Array wo jede Zeile der Tabelle repräsentiert (Beliebt in C)
- Zustandsobjekte
 - Jeder Zustand \triangleq 1 Objekt
 - Java: Enum, die Zustände repräsentiert
 - + Methoden, die Verhalten in jeweiligen Zustand repräs.
 - + Methoden, die in neuen Zustand wechseln

WIE IMPLEMENTIEREN?

```

public class Person{
    FamilienStand _fs= FamilienStand.ledig;
    Person _ehePartner= null;

    public enum FamilienStand implements ... {
        ledig(...), verheiratet(...), geschieden(...),
        verwitwet(...), tot;

        private FamilienStand(...) { ... }

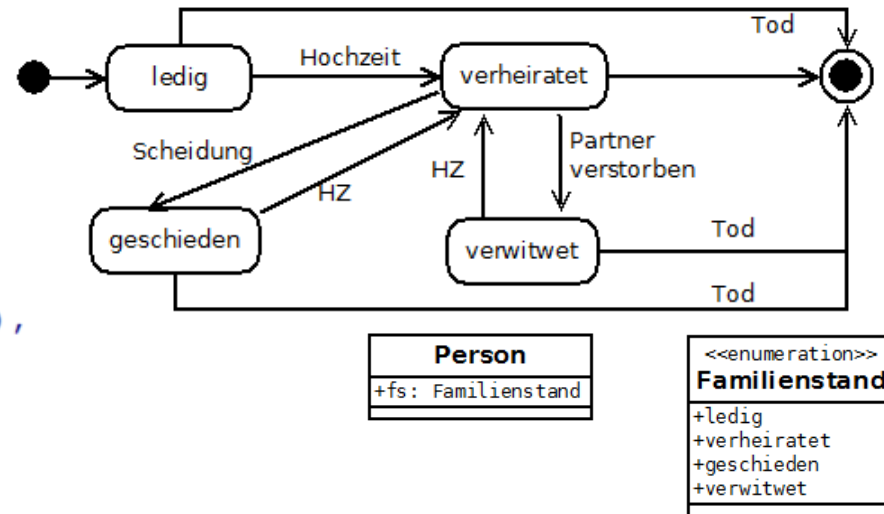
        public boolean kannHeiraten(){
            return this == ledig || this == geschieden || this == verwitwet;
        }

        public void heirate(Person partner){
            if(this.kannHeiraten() && partner._fs.kannHeiraten()){
                Person.this._ehePartner= partner;
                partner._ehePartner= Person.this;
                Person.this._fs= verheiratet;
                partner._fs= verheiratet;
            }
            throw new InvalidStateChangeException("...");
        }

        public void scheide(){
            ErrorHandler.Assert(Person.this._fs == verheiratet, ... );
            ErrorHandler.Assert(Person.this._ehePartner._fs == verheiratet, ... );
            ErrorHandler.Assert(Person.this._ehePartner._ehePartner == Person.this, ... );

            Person.this._fs == geschieden;
            Person.this._ehePartner._fs == geschieden;
            Person.this._ehePartner._ehePartner= null;
            Person.this._ehePartner= null;
        }
    }
}

```



WIE IMPLEMENTIEREN?

```
public class Person{
    FamilienStand _fs= FamilienStand.ledig;
    Person _ehePartner= null;

    public enum FamilienStand implements ... {
        ledig(...), verheiratet(...), geschieden(...),
        verwitwet(...), tot;

        private FamilienStand(...) { ... }
        ...
    }

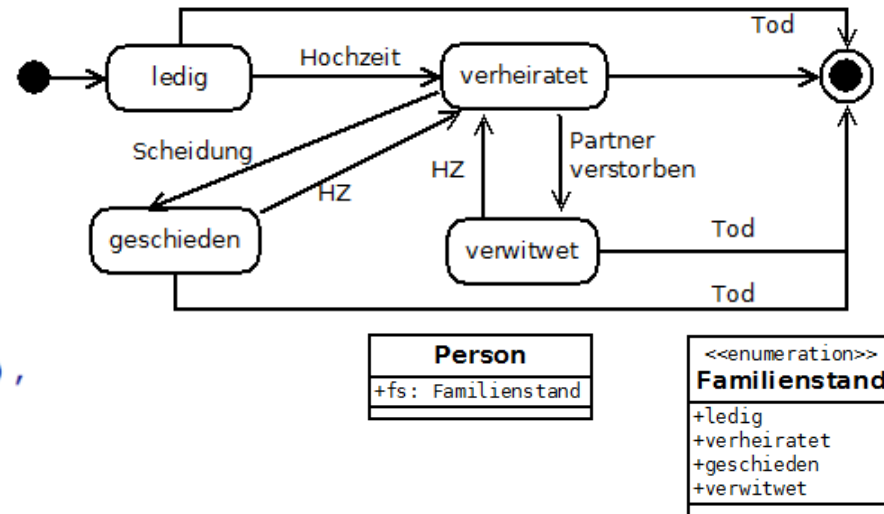
    public void sterbe(){
        Person.this._fs= tot;

        if(Person.this._ehePartner != null){
            ErrorHandler.Assert(Person.this._ehePartner._fs == verheiratet, ... );
            Person.this._ehePartner._fs= verwitwet;
        }
    }

    public void heirate(Person partner){
        this._fs.heirate(partner);
    }

    public void scheide(){
        this._fs.scheide();
    }

    public void sterbe(){
        this._fs.sterbe();
    }
}
```



Ähnliche Wirkweise wie
Method-Object-Pattern (PMT (2. Sem))

- Komplexität wird in innerer Klasse Gekapselt
- Hier aber die Komplexität des Zustandsautomaten



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

05 Fazit

Ziel:
Was haben wir damit gewonnen?



WAS HABEN WIR GELERNT?



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

- Zustandsdiagramm
- Wie man Zustände und dessen Übergänge modelliert
- Alternative Notation via Tabelle
- Sonstige Informationen



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

AUF GEHT'S!!

SELBER MACHEN UND LERNEN!!

