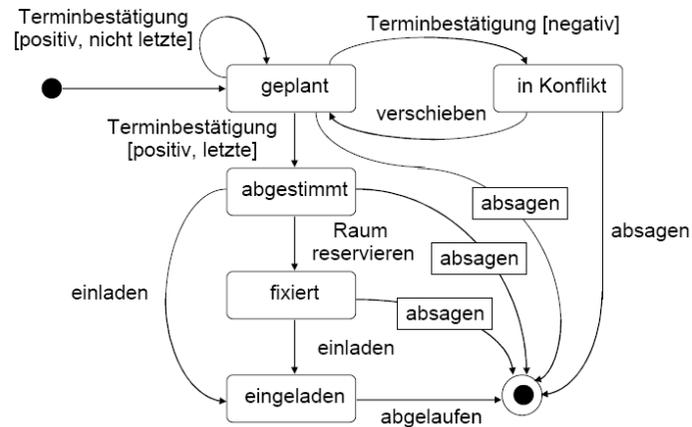


Einführung: Zustandsdiagramme Stand: 01.06.2006

Josef Hübl (Triple-S GmbH)



1. Grundlagen

Zustände, Ereignisse, Bedingungen, Aktionen

2. Verkürzte Darstellungen

Pseudozustände

3. Hierarchische Zustandsgraphen

Superzustände, Teildiagramme

Definitionen

- Ein Zustandsdiagramm ist ein Graph mit Zuständen als Knoten und Zustandsübergängen als Kanten
- Events/Ereignisse sind Zustandsübergänge in anderen Zustandsmaschinen/Zustandsdiagrammen
- Eine Bedingung liefert den logischen Wert TRUE oder FALSE

Voraussetzungen

- Die Zustandsvariable kann nur eine endliche Anzahl (<10), aber mindestens 2 verschiedene Werte (Zustände) annehmen.
- Die Zustandsmaschine speichert ihren Zustand, während sie auf Ereignisse wartet.
- Die Zustandsvariable ist immer lokal gespeichert und darf von außen nicht verändert werden.
- Die Zustandsvariable darf von innen nur wie in Zustandsübergängen definiert verändert werden.
- Es gibt nur einen Übergang von einem Zustand in einen anderen . (Keine Mehrfachverbindungen!)

Graphische Darstellung



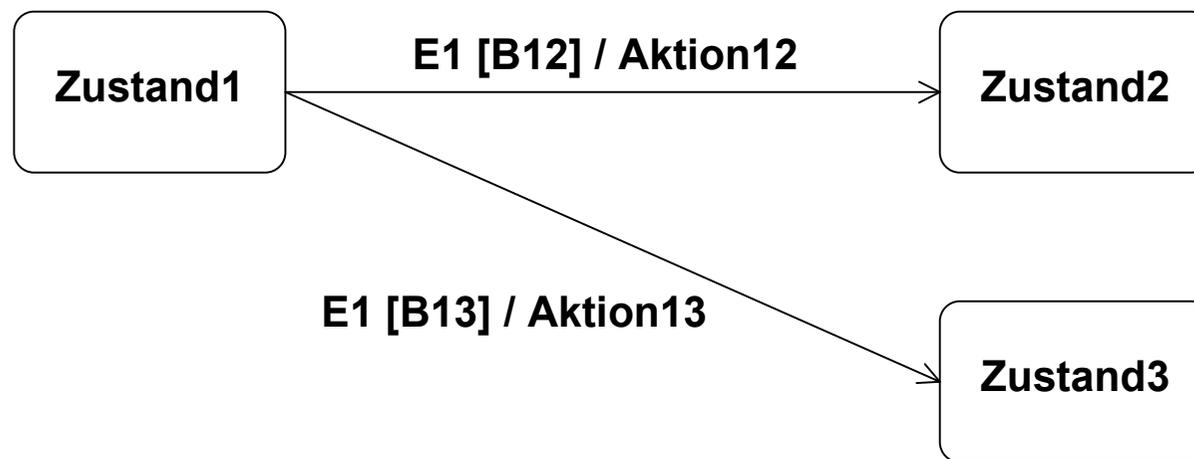
Wenn sich die Zustandsmaschine im Zustand1 befindet und das Ereignis eintritt und die Bedingung erfüllt ist, dann wird die Aktion ausgeführt und in den Zustand2 gewechselt.

Alle Bestandteile sind optional.

Ein Übergang ohne Ereignis oder Bedingung erfolgt immer, sobald der Zustand1 erreicht wird.

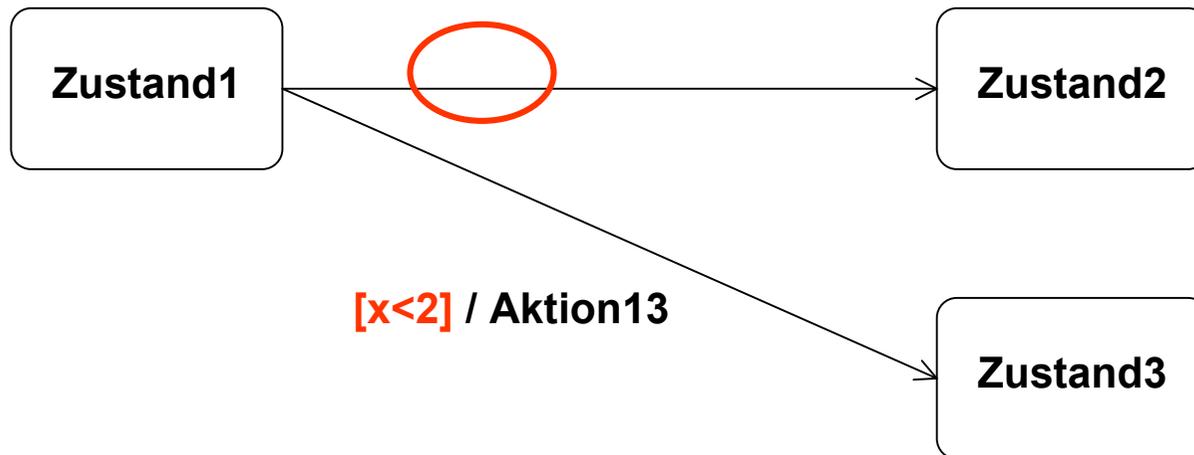
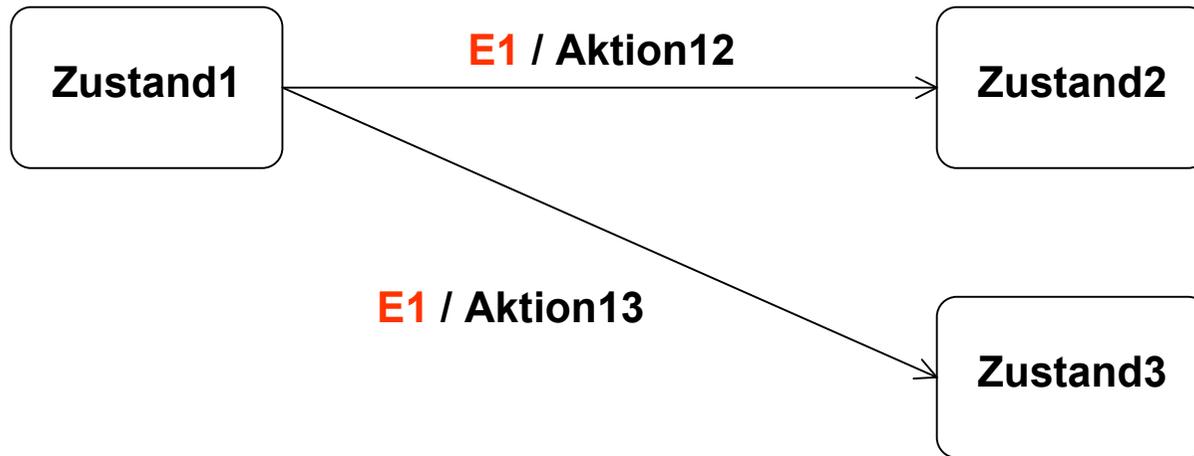
Eine Aktion muss nicht sein.

Eindeutigkeit



Über Bedingungen ist zu steuern, dass von einem bestimmten Zustand ein Ereignis nicht gleichzeitig zu verschiedenen Nachfolgezuständen führt, sonst implementierungsabhängiges Verhalten!

Fehler



Kapselung



Eine Aktion (intern oder extern) darf den Wert der Zustandsvariablen nicht verändern!

Mögliche Umsetzung in Sourcecode



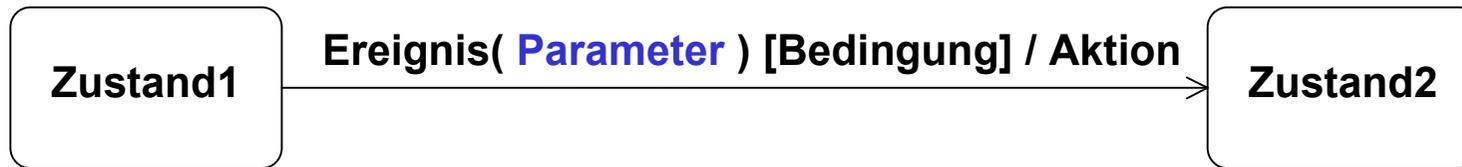
```
void Ereignis( void )
{
    switch ( zustand )
    {
    case Zustand1:
        if ( Bedingung )
            { Aktion; zustand = Zustand2; }
        break;
    }
}
```

Designfehler!

```
switch( zustand )
{
case Zustand1:
    switch ( ereignis )
    {
case Ereignis1:
        if ( Bedingung )
            { Aktion; zustand = Zustand2; }
        break;
    }
    break;
}
```

Ereignisse entsprechen Aufruf von Funktionen bzw. Methoden eines Objekts!

Ereignis als Funktionsaufruf



UML erlaubt Angabe von Parametern bei Ereignis die zu Beschreibung von Bedingungen und Aktionen verwendet werden können.

Was fehlt für Sourcecode?

- **Keine Typdeklarationen für verwendete Variablen (außer Zustandsvariable - Enumeration)**
- **Keine Gültigkeitsbereiche für verwendete Variablen (außer Zustandsvariable - nur lokal)**
- **Offen ist, wie Ereignisse in Funktionsaufrufe umgewandelt werden!**

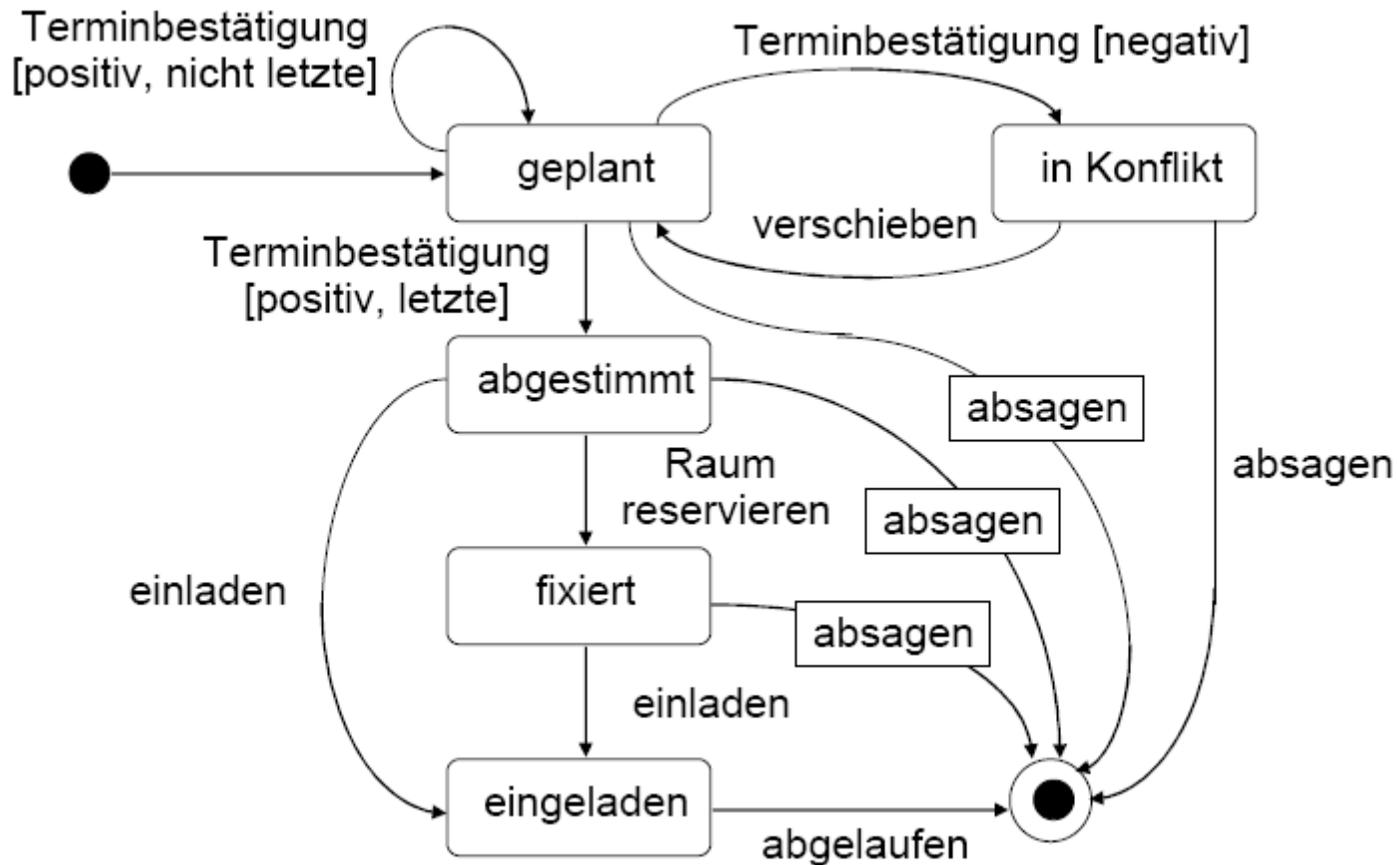
Lebenszeit: Anfang und Ende



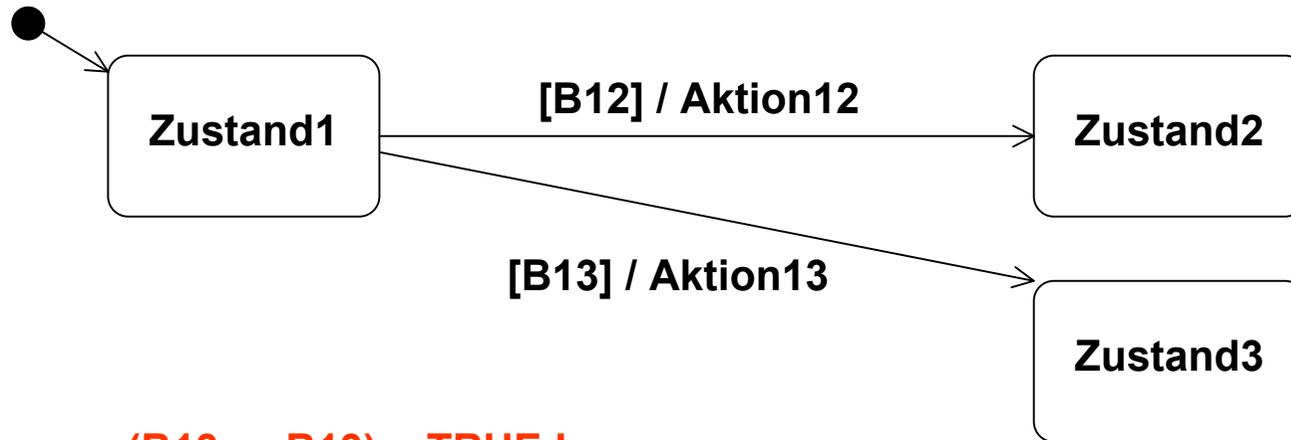
Initialzustand
nach Instanziierung

Terminalzustand
anschließende Zerstörung

Beispiel Terminplanung

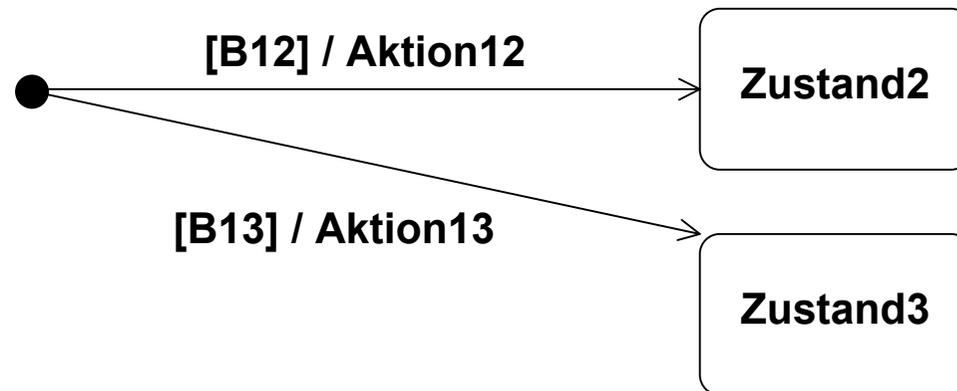


Verkürzte Darstellungen

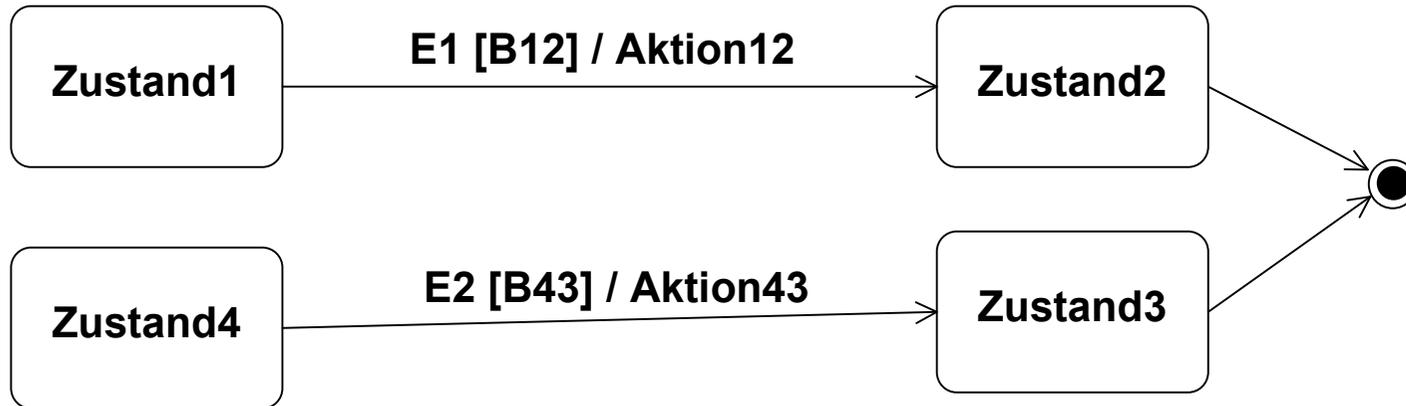


Voraussetzung: (B13 or B13) = TRUE !

Kurzform:



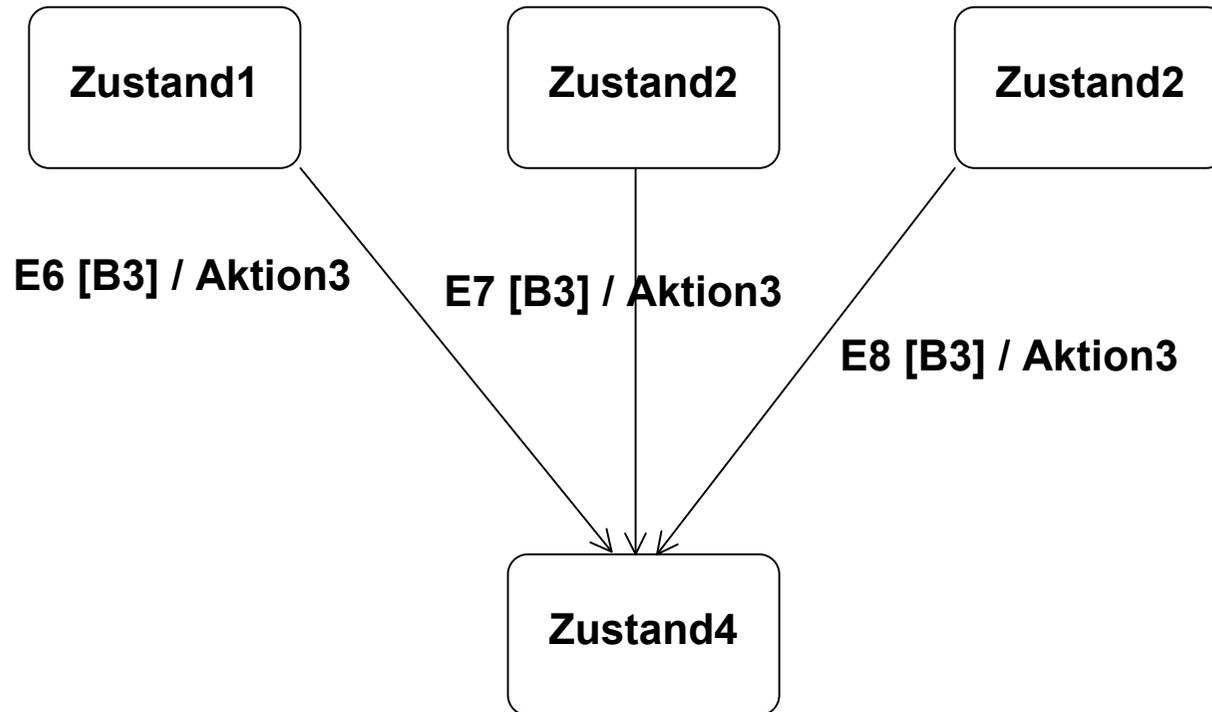
Verkürzte Darstellungen



Kurzform:

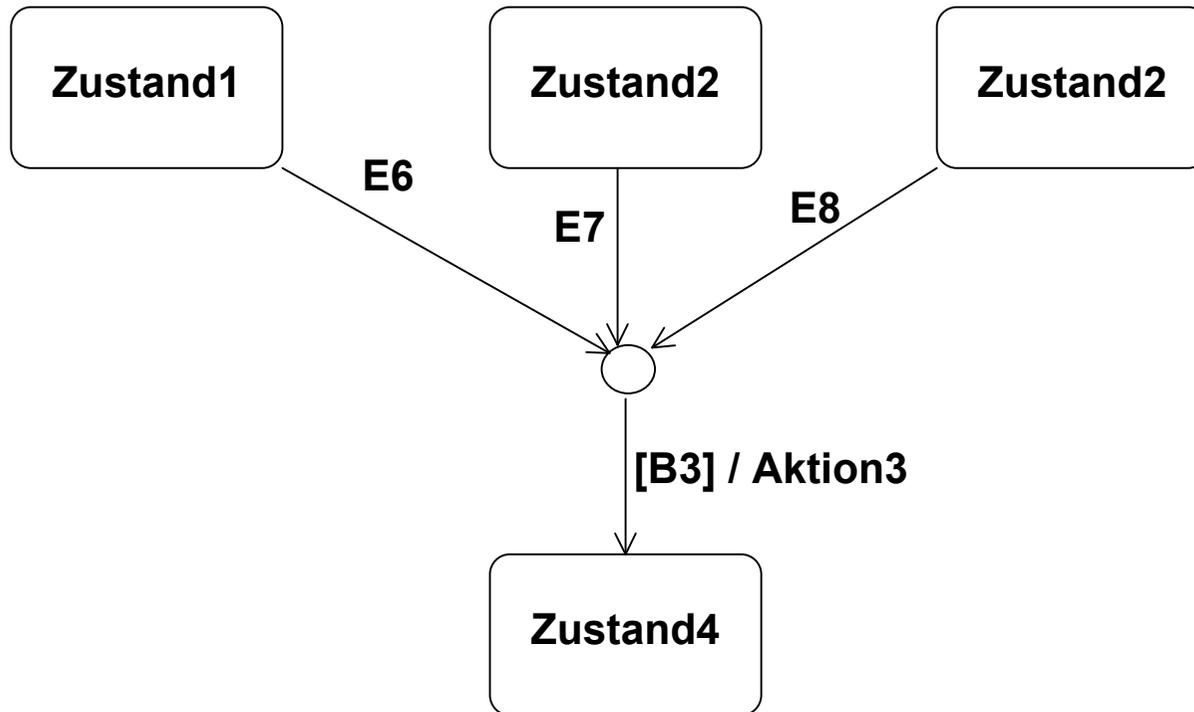


Verkürzte Darstellungen



Gleiche Bedingung und gleiche Aktion !

Verkürzte Darstellungen

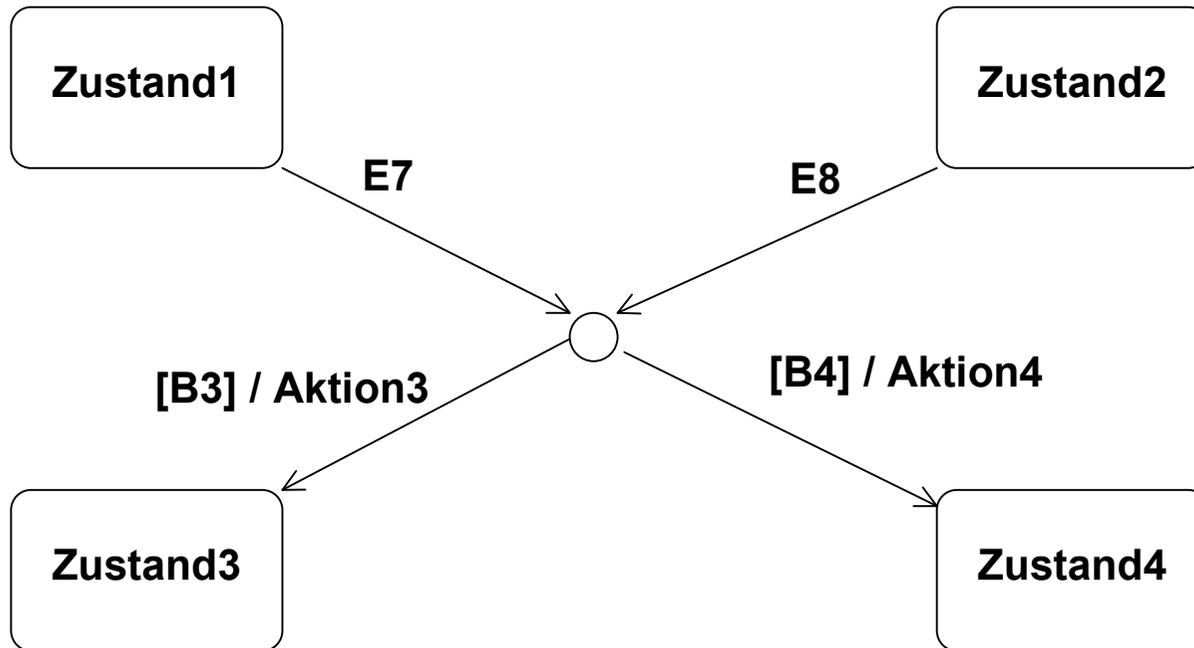


Kurzform:

Nach Pseudozustand keine Abfrage auf Ereignis !

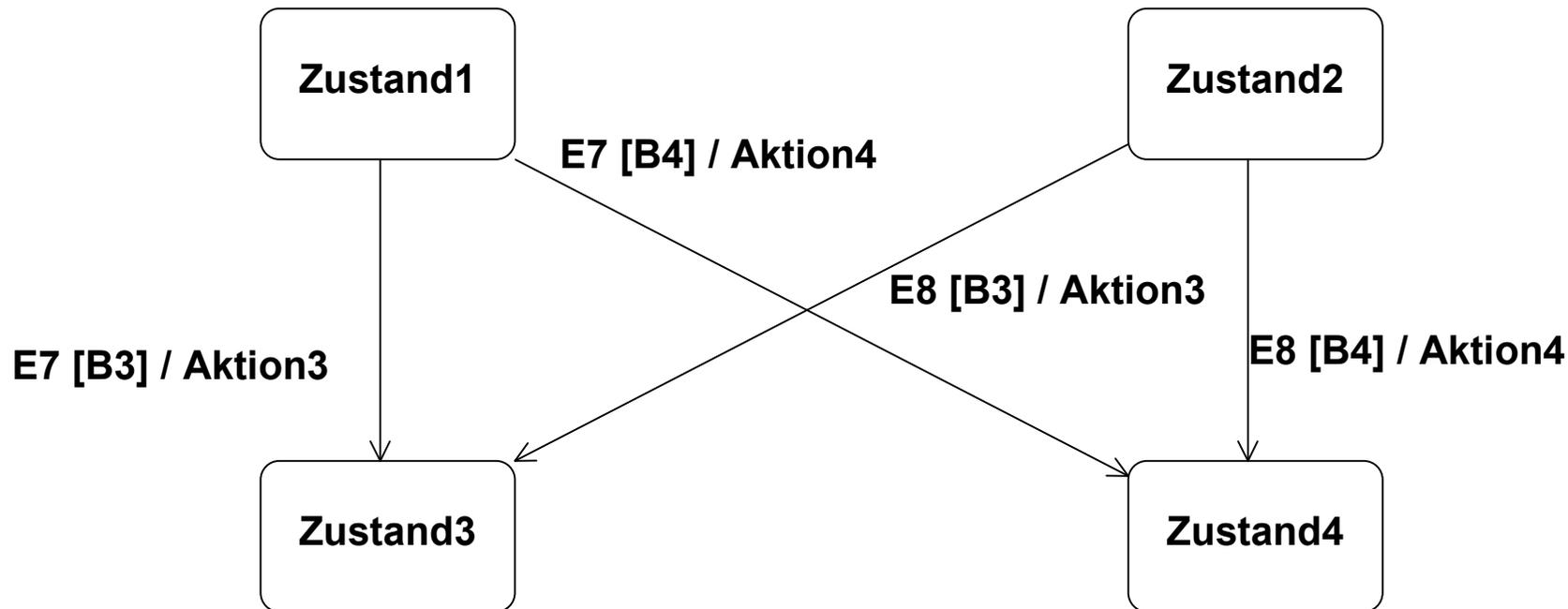
Verkürzte Darstellungen

Kurzform:



Nach Pseudozustand keine Abfrage auf Ereignis !

Komplexe Darstellung



Jeder Übergang von einem Zustand vor den Pseudozustand zu jedem Zustand nach den Pseudozustand mit gleicher Bedingung und Aktion, aber verschiedenen Ereignissen!

Verkürzte Darstellungen

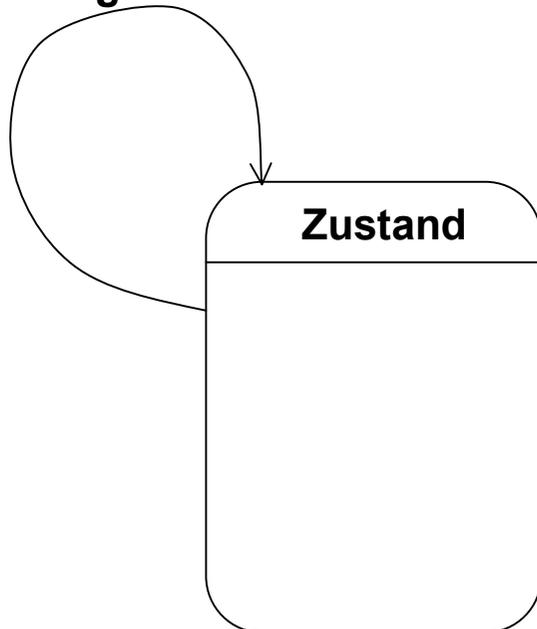


Entry: Aktion immer, wenn in den Zustand gewechselt wird, unabhängig davon aus welchem anderen Zustand.

Exit: Aktion immer, wenn der Zustand verlassen wird, unabhängig davon aus welcher Ursache.

Verkürzte Darstellungen

Ereignis / Aktion



Kurzform:

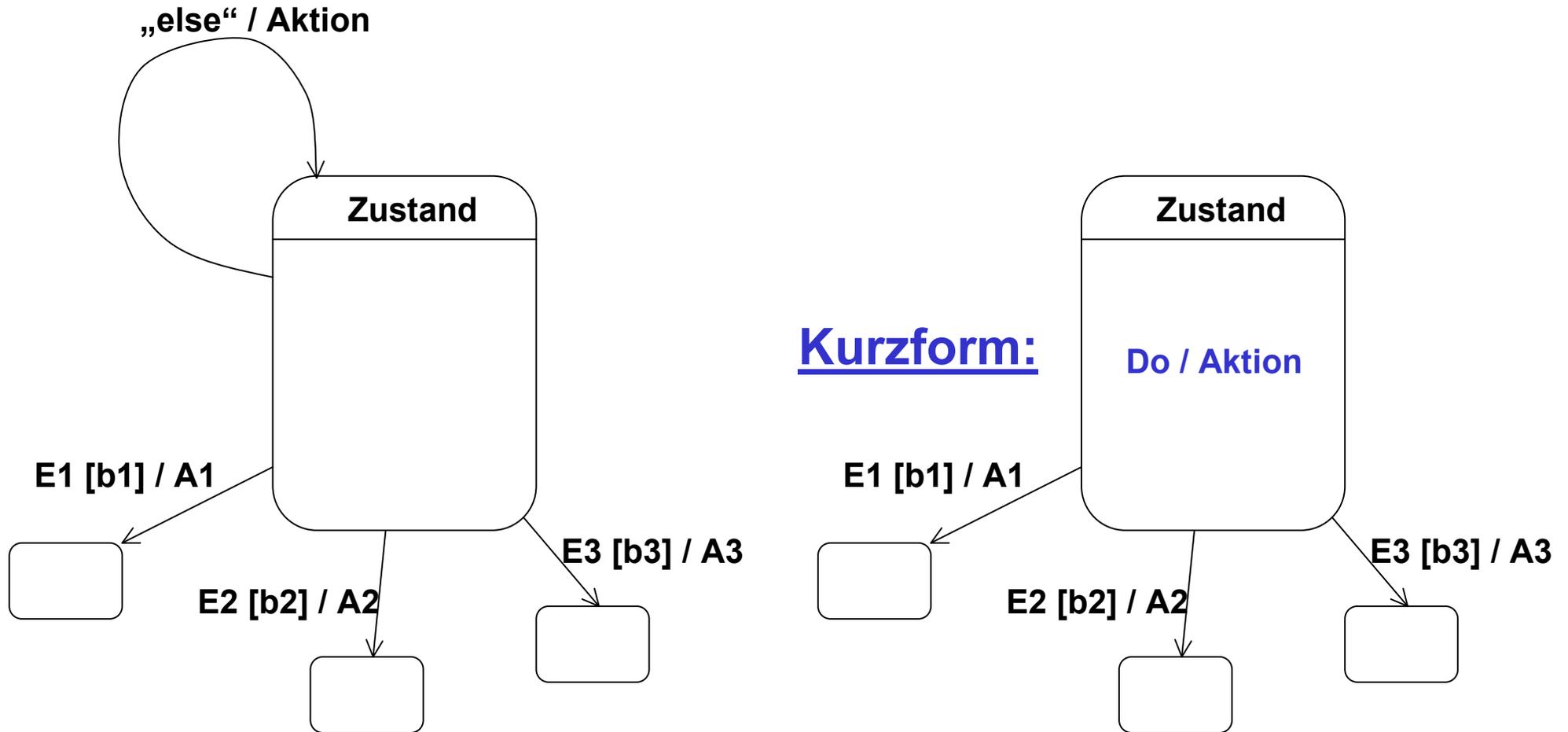


“else“ Do

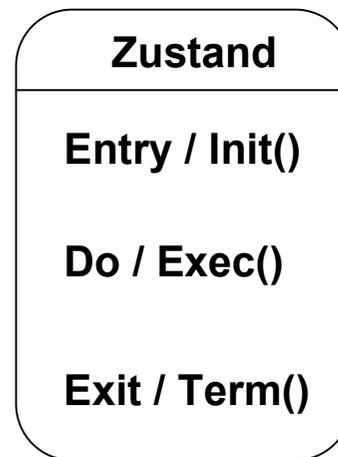


Do: Aktion wird bei Wechsel in Zustand begonnen und solange ausgeführt, bis der Zustand verlassen wird.

Verkürzte Darstellungen

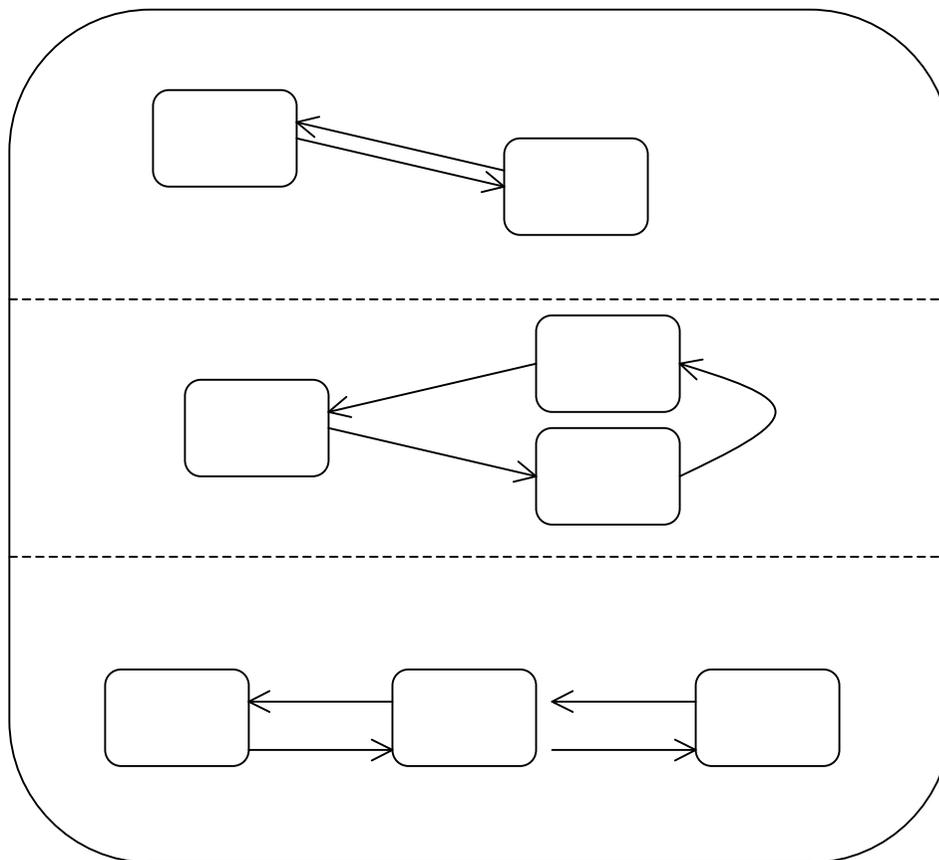


Parallele Aktivitäten



**Eigenständiges Objekt wird gebunden an den Zustand
kreiert, initialisiert, ausgeführt und terminiert.**

Parallele Zustandsmaschinen



**Zustände sind
orthogonale Produkte
mehrerer Zustands-
variablen.**

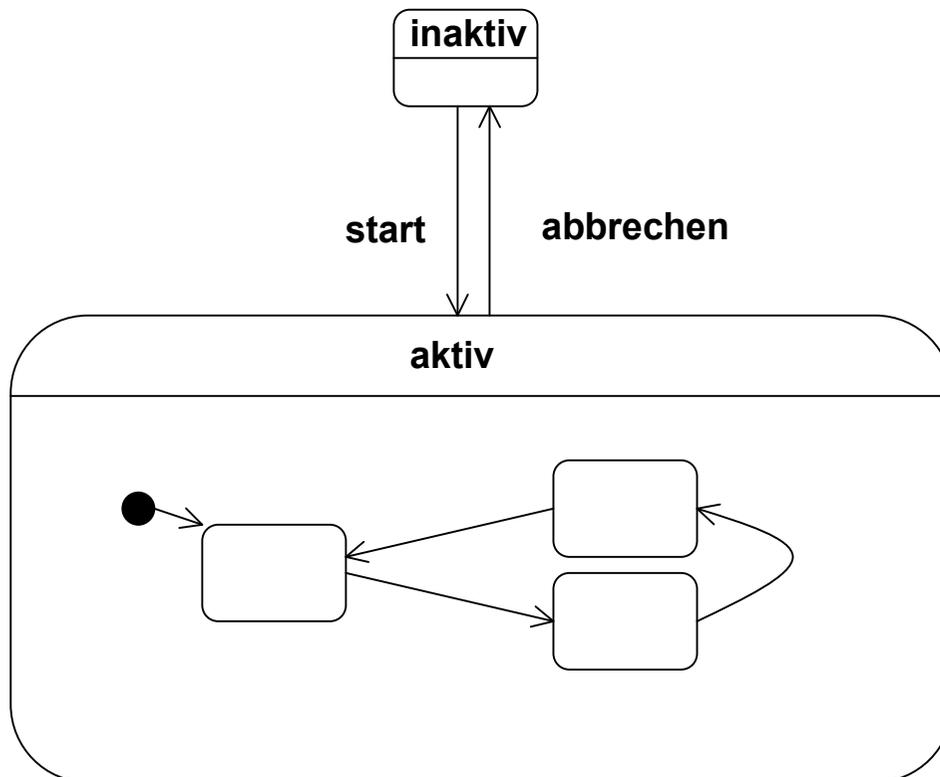
z.B. (Z1a, Z2b, Z3x)

**Übergänge finden
gleichzeitig statt!**

**(Besondere Implementierung für
gleichzeitige Zustandsübergänge!)**

**Sie beeinflussen sich
gegenseitig!**

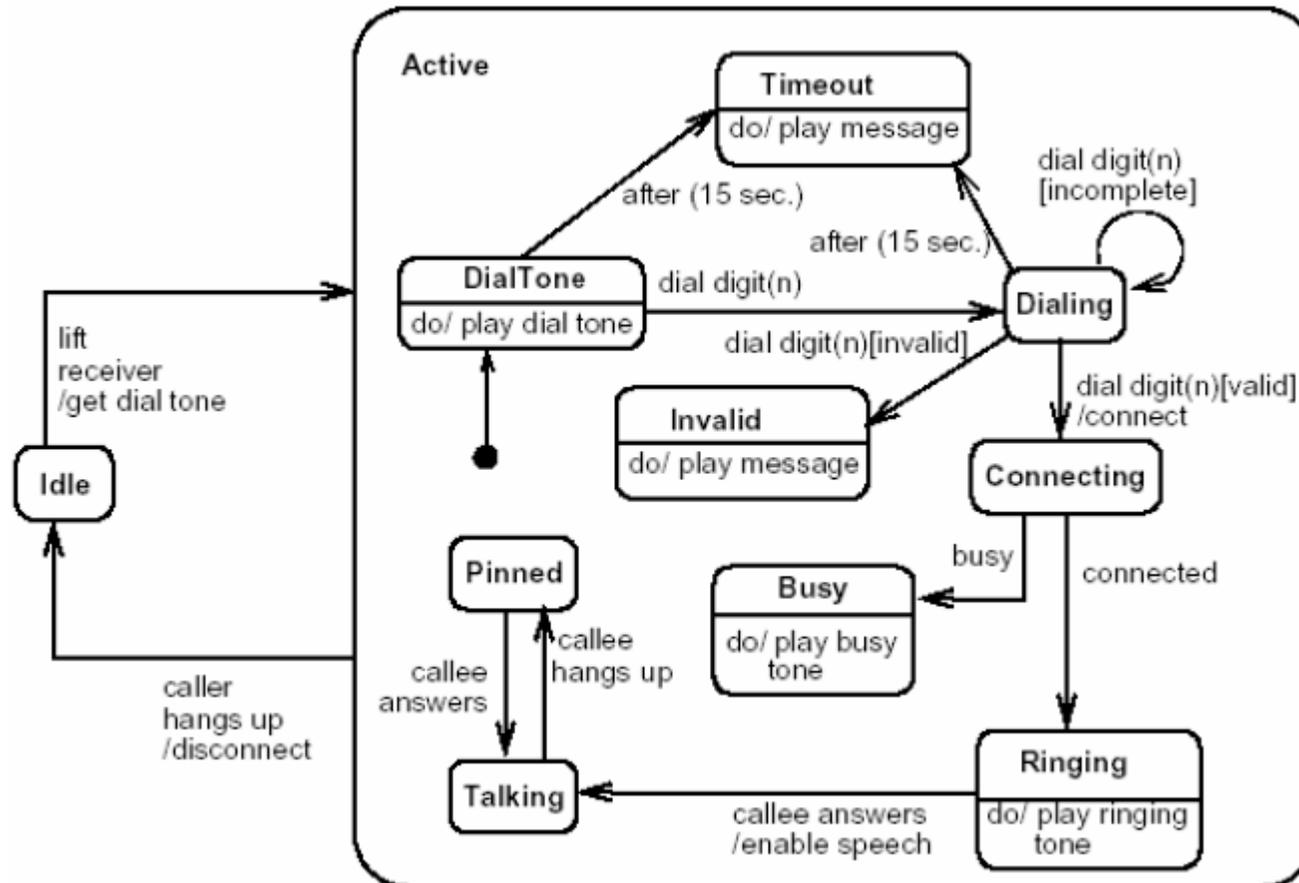
Superzustände



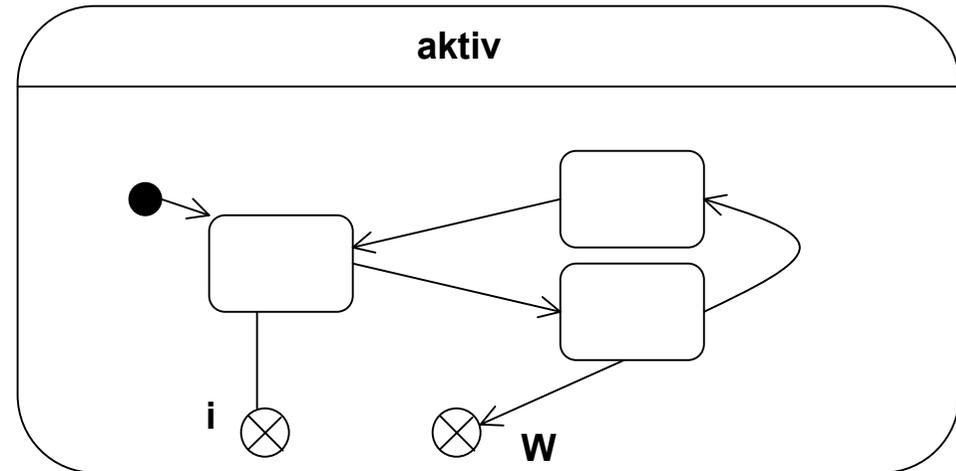
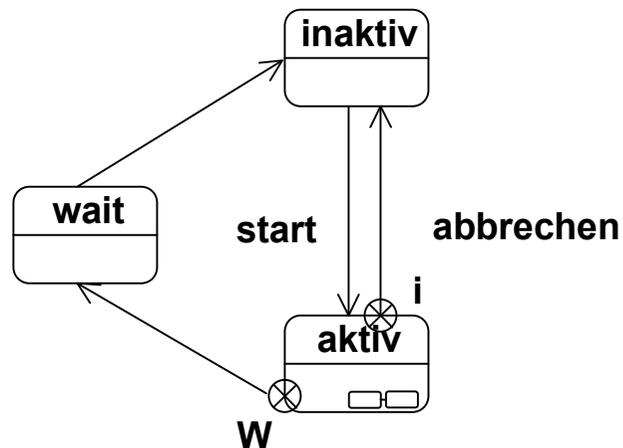
Übergang aus dem Superzustand gilt für jeden Unterzustand.

Übergang in den Superzustand gilt für Initialzustand.

Beispiel Telefonvermittlung



Teildiagramme

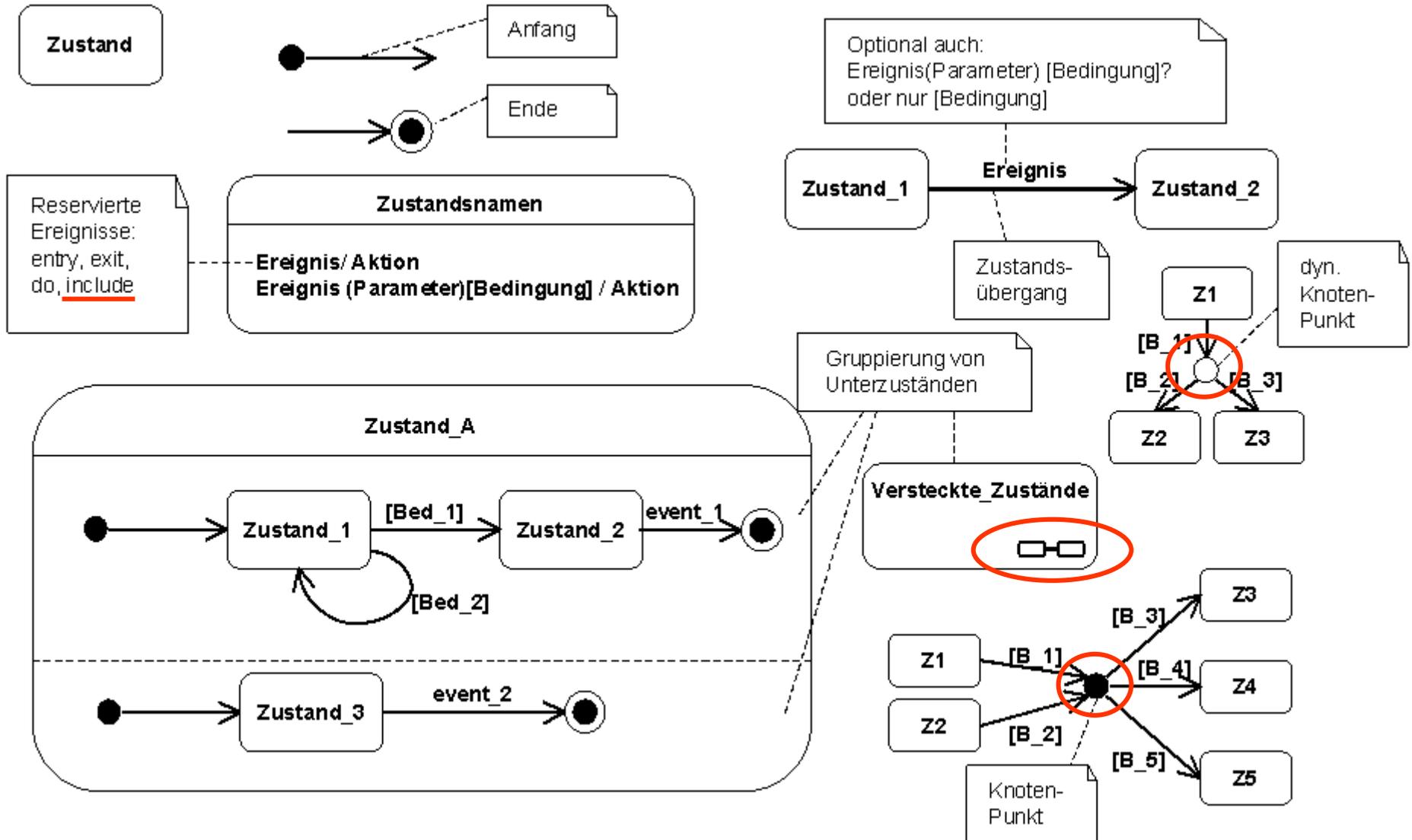


Übergang aus dem Teildiagramm kann für jeden Unterzustand festgelegt werden.

Übergangsknoten müssen benannt sein!

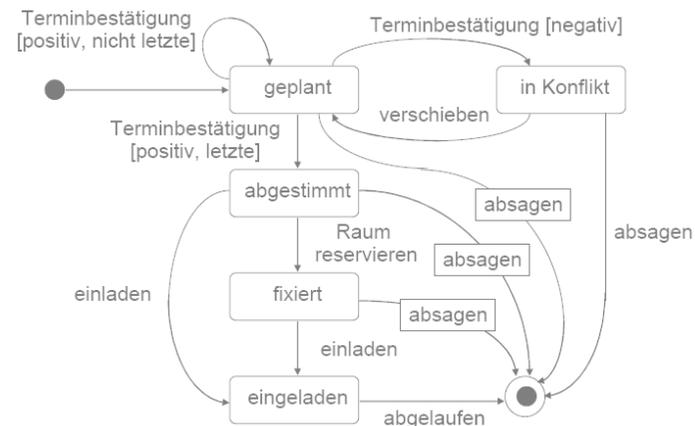
Zustandsdiagramme

UML



Einführung: Zustandsdiagramme Stand: 05.04.2006

Josef Hübl (Triple-S GmbH)



Danke für Ihre Aufmerksamkeit!