

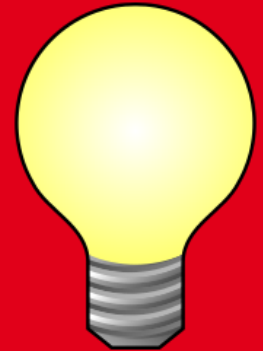


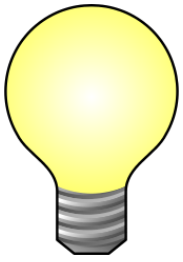
Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

17.12.2020

Interaktionsdiagramme

Interaktionsdiagramme nutzen





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

AGENDA

Einführung ins Thema

Sequenzdiagramme

Sequenzdiagramme – 2 verschiedene Semantiken

Kommunikationsdiagramme

Weitere Interaktionsdiagramme

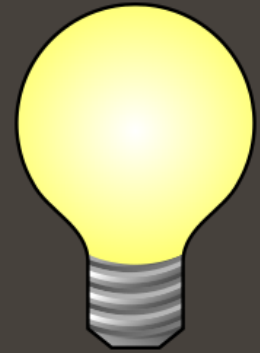
Fazit



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

01 EINFÜHRUNG INS THEMA

Ziel:
Die Eckpunkte des Themas kennenlernen

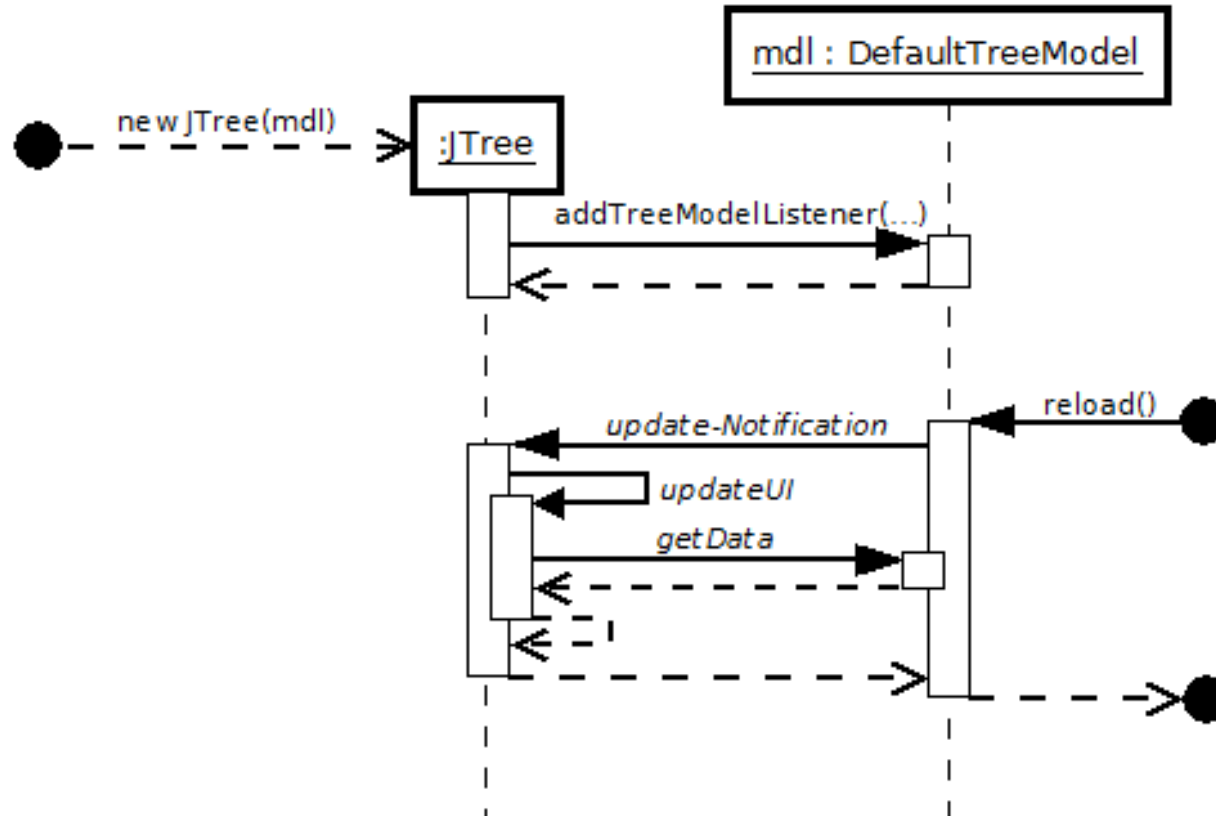


2. SEMESTER, PM: DEN JTREE VERÄNDERN – HINTERGRÜNDE VON RELOAD()



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

Sequenz-Diagramm über den groben Ablauf:



Hinweis:

Kursiv gesetzten Ausdrücke *update-Notification*, *updateUI* und *getData* zeigen nur ungefähr den Ablauf, die wahren Namen der Funktionen sind Javainternas!

INTERAKTIONSDIAGRAMME



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

- Zeigen die Interaktionen zwischen Objekten
 - Interaktion = Abfolge von Nachrichten, die zwischen Objekten ausgetauscht werden
- Grundsätzlich gibt es 4 Interaktionsdiagramme in UML:
 - Sequenzdiagramm (wichtigste)
 - Kommunikationsdiagramm
 - Interaktionsübersichtsdiagramm
 - Metadiagramm, das die Zusammenhänge zw. verschied. Interaktionsdiagrammen zeigt
 - Timing Diagramm:
 - Zeigt Nachrichtenaustausch und Zustandwechsel verschiedener Objekte zu bestimmten Zeitpunkten an.

} Diese besprechen wir
im Detail

BEVORZUGTE EINSATZGEBIETE



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

- Man hat mehrere Einheiten (Objekte), die Nachrichten austauschen
- Zeitliche Abfolge der Nachrichten soll dargestellt werden
- Snapshot-Charakter (ähnlich wie Objektdiagramme)



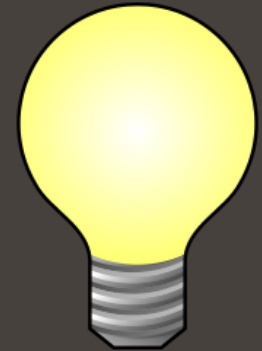
Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

02

Sequenzdiagramme - Modellelemente im Überblick

Ziel:

Elemente im Überblick erfassen



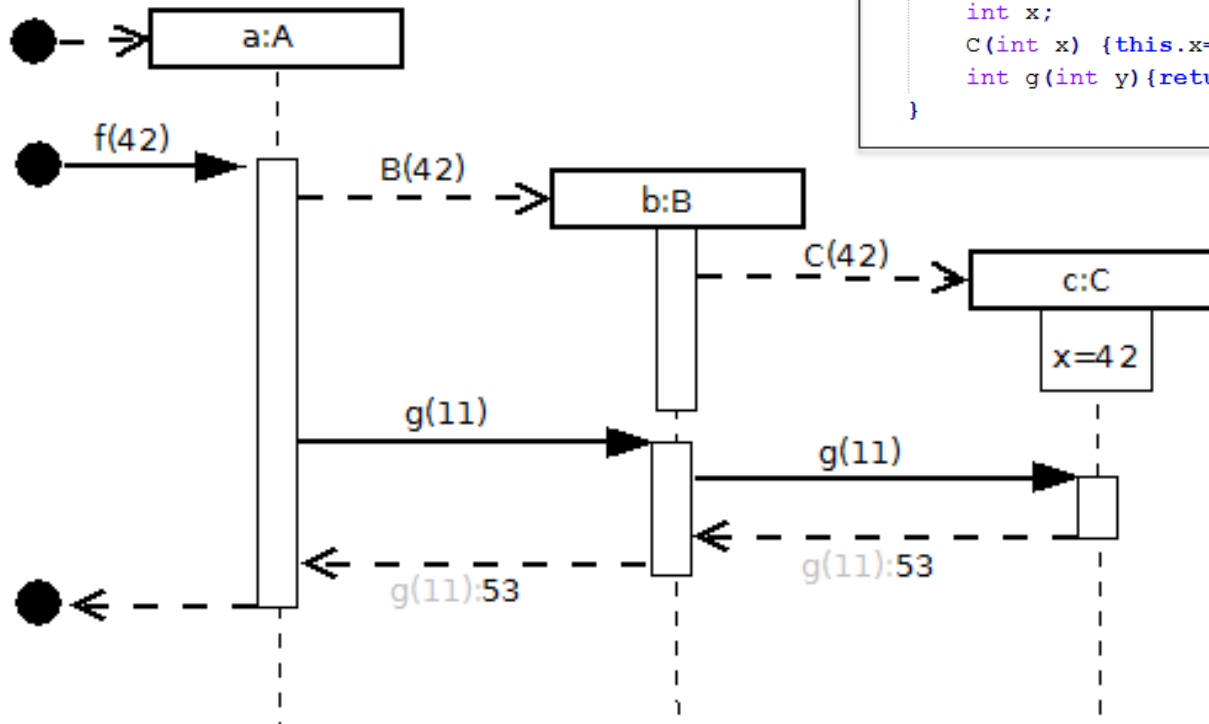
CODE-BEISPIEL:



```
class A {  
    void f(int x){  
        B b = new B(x);  
        int z;  
        z=b.g(11);  
    }  
  
    public static void main(String [] args){  
        A a=new A();  
        a.f(42);  
    }  
}  
  
class B{  
    C c;  
    B (int x) { c=new C(x);}  
    int g(int y){return c.g(y);}  
}  
  
class C {  
    int x;  
    C(int x) {this.x=x;}  
    int g(int y){return x+y;}  
}
```


DER CODE ALS SEQUENZDIAGRAMM:

```
class A {  
    void f(int x){  
        B b = new B(x);  
        int z;  
        z=b.g(11);  
    }  
  
    public static void main(String [] args){  
        A a=new A();  
        a.f(42);  
    }  
}  
  
class B{  
    C c;  
    B (int x) { c=new C(x);}  
    int g(int y){return c.g(y);}  
}  
  
class C {  
    int x;  
    C(int x) {this.x=x;}  
    int g(int y){return x+y;}  
}
```

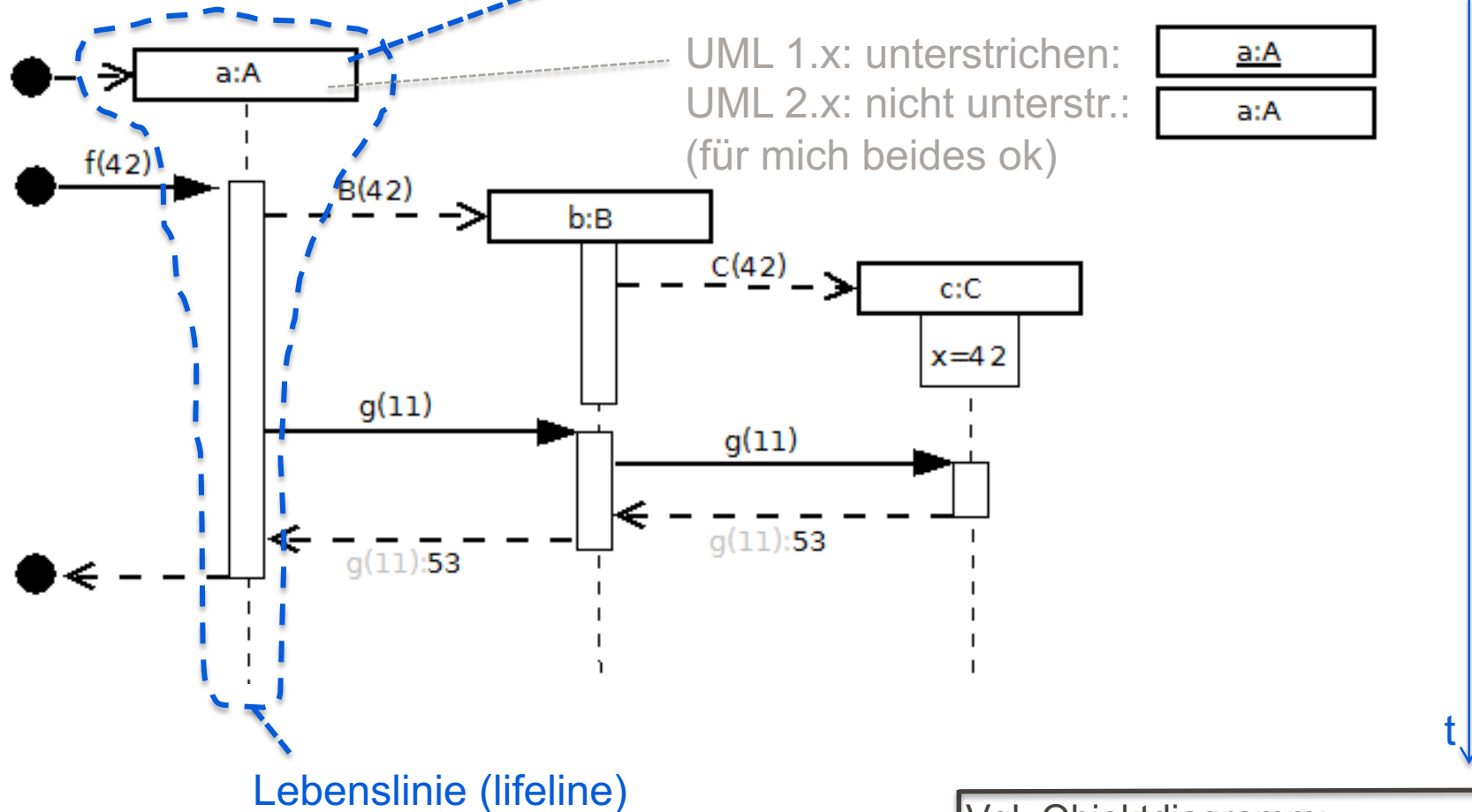


DIE ELEMENTE IM DETAIL:

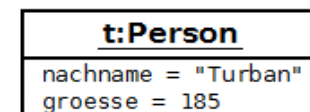


Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

Objekt /
Kommunikationspartner



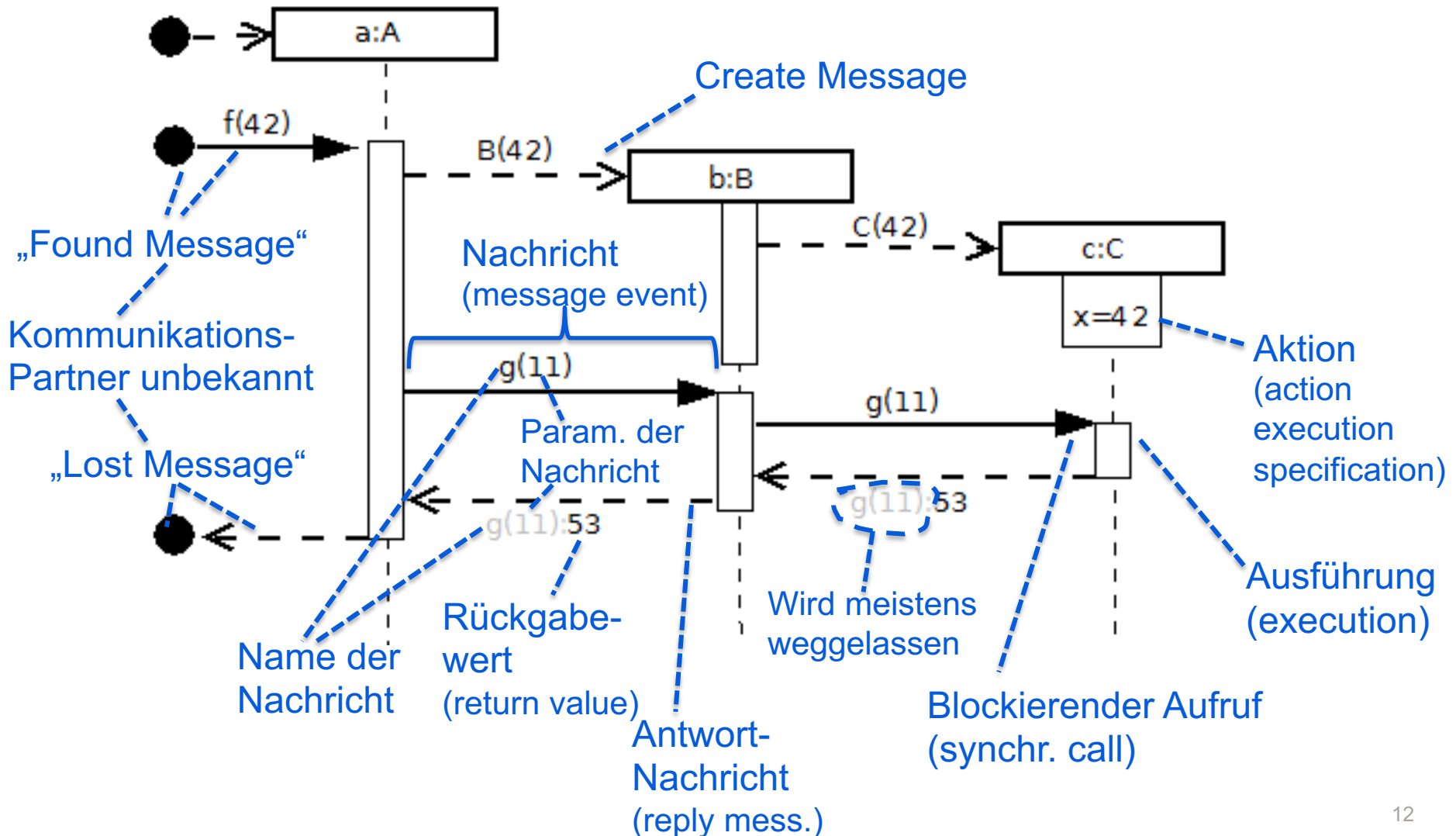
Vgl. Objektdiagramm:



DIE ELEMENTE IM DETAIL:



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



GENERELL MÖGLICHE NACHRICHTENTYPEN:



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

1. Erzeugungsnachricht (Create Message):

2. Synchrone Nachricht (blockiert):



- Sender wartet, bis Empfänger die Nachricht abgearbeitet hat
- Gestrichelter Pfeil für Rücksprung zum Sender nötig!

3. Asynchrone Nachricht (blockiert nicht):



- Sender wartet nicht auf Empfänger und arbeitet unmittelbar weiter
→ Sender und Empfänger befinden sich in unterschiedlichen Ausführungsprozessen
- Kein gestrichelter Rückgabepfeil!!

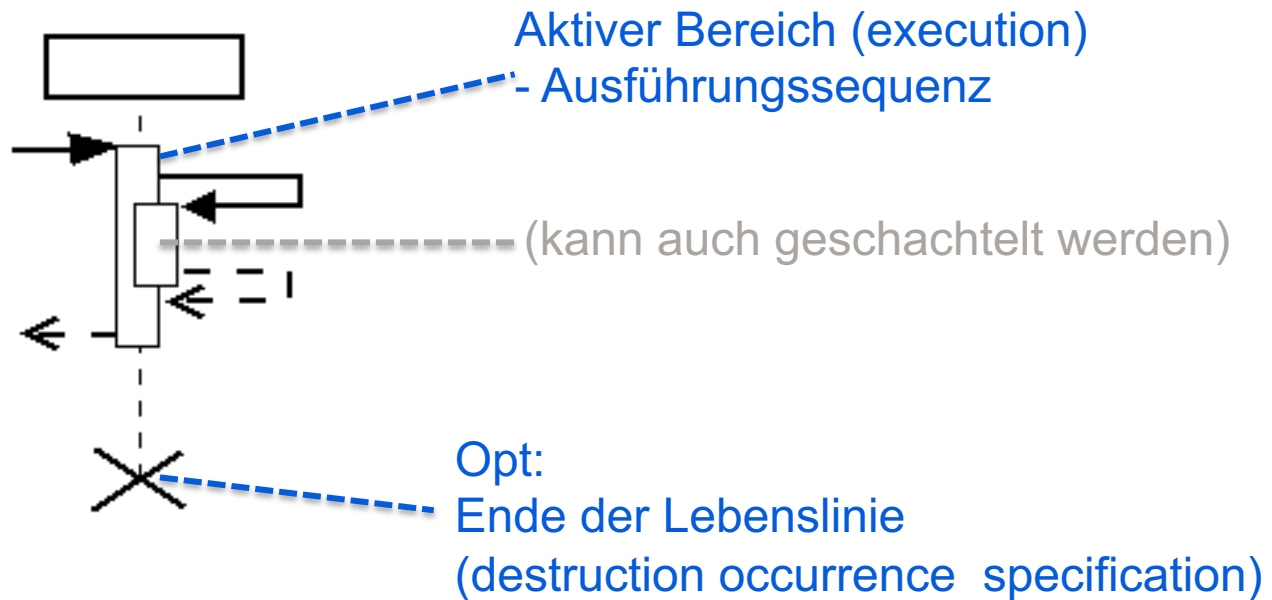
BEM: Ausnahmen (Exceptions) werden auch “nur” als normale Nachrichten behandelt.

DIE ELEMENTE IM DETAIL:



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

Die Lebenslinie im Detail:

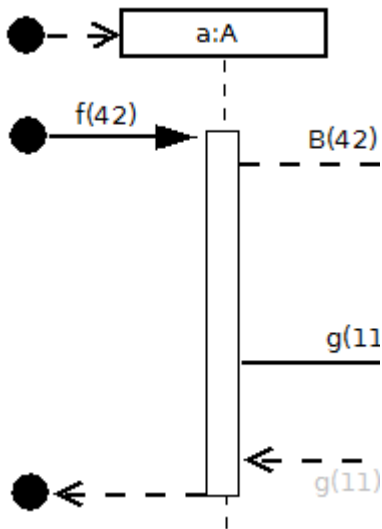


LOST-FOUND MESSAGES VS SOG. GATES

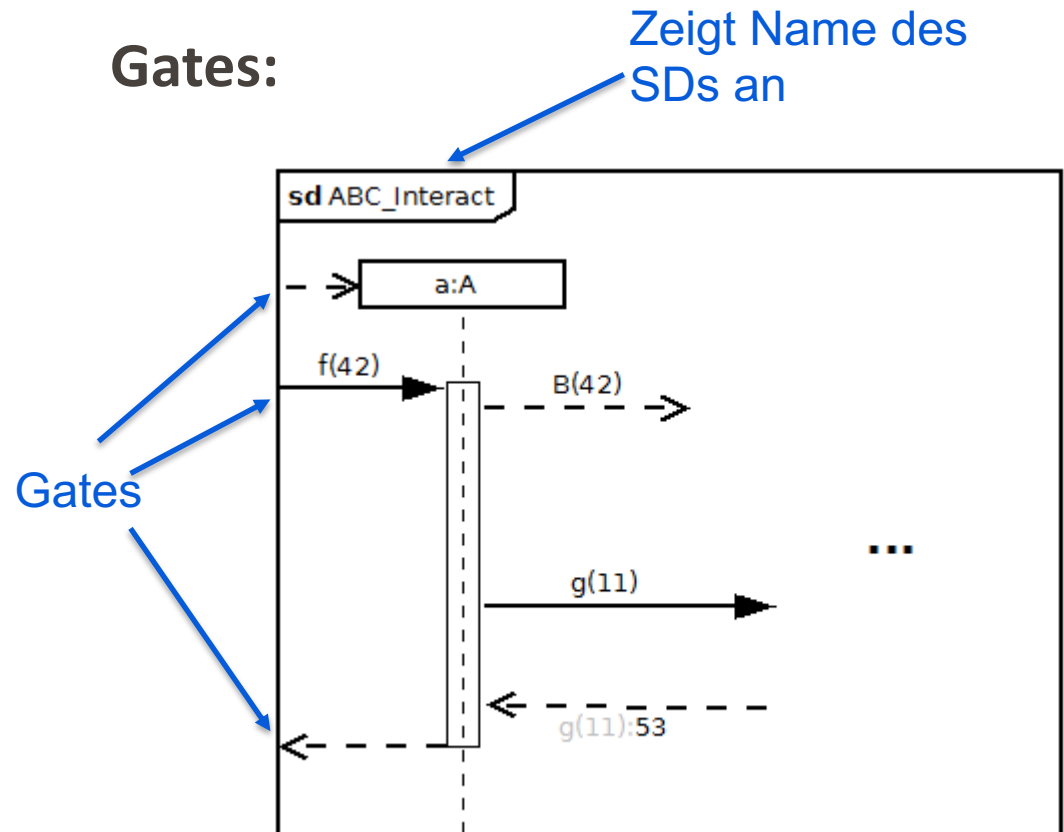


Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

Lost-Found-Messages:



Gates:

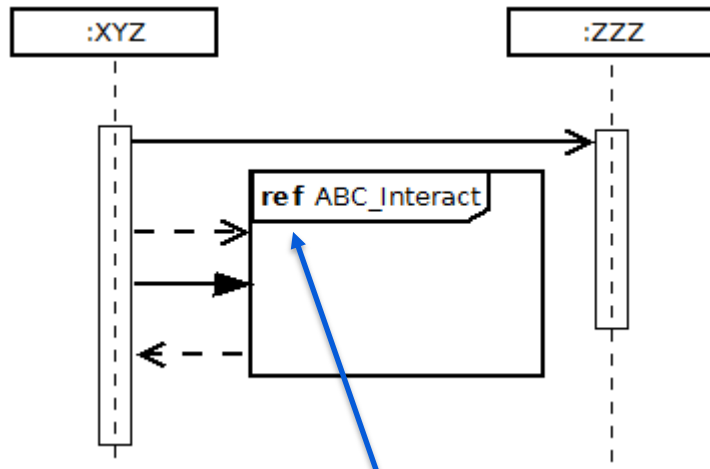


BEM LOST-FOUND MESSAGES UND SOG. GATES

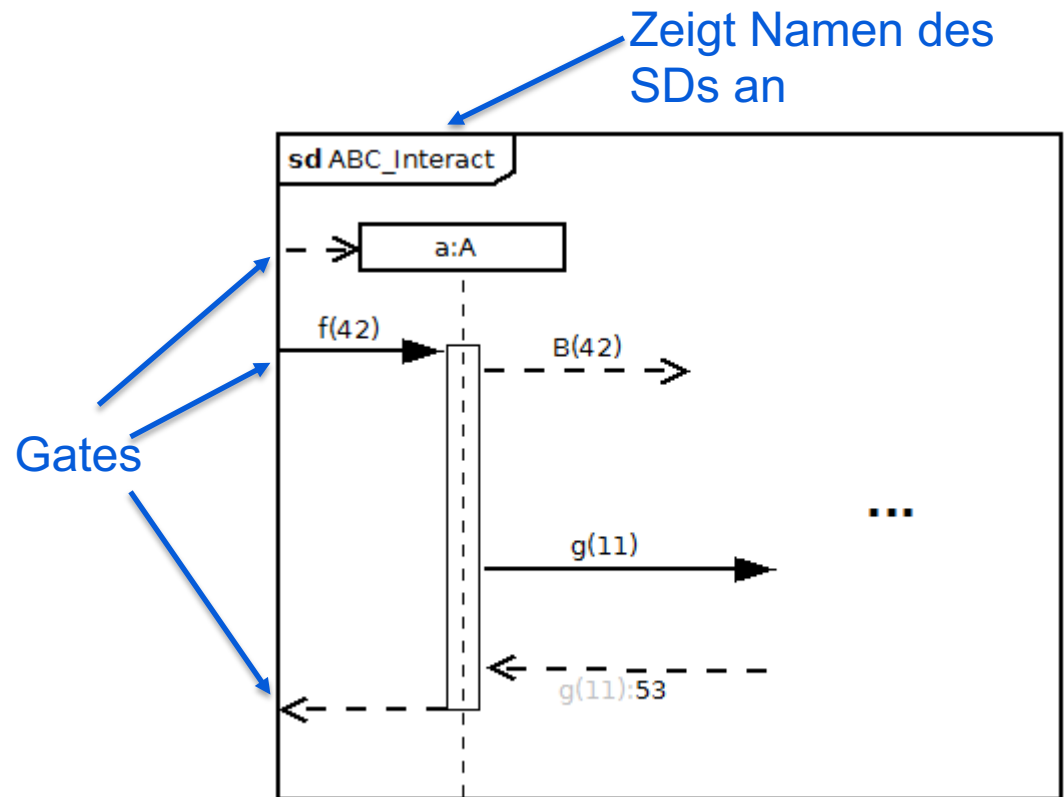


Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

→ Gates sind Anknüpfungspunkte, die bei Referenzen aufgenommen werden:



„ref“ zeigt an,
dass anderes
Diagramm
referenziert wird



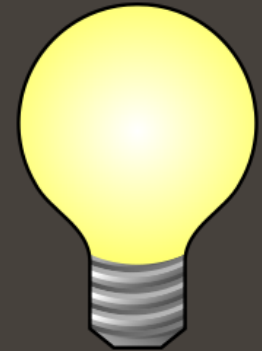


Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

03

Sequenzdiagramme – 2 verschiedene Semantiken

Ziel:
Elemente im Überblick erfassen



ZWEI VERSCHIEDENE SEMANTIKEN



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

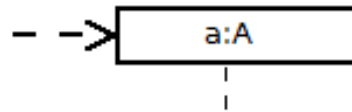
- Die wichtigsten Elemente des Sequenzdiagramms kennengelernt
 - Syntax
- Es gibt aber auch zwei Arten der Bedeutung (Semantik):
 - "Aufrufsemantik"
 - Orientiert sich daran wie in Programmiersprachen Methoden aufgerufen werden
 - Sehr nah an Programmiersprachen
 - "Signalsemantik"
 - Sagt nur allgemein aus, dass "Signale ausgetauscht" werden
 - Viel allgemeinere/unbestimmte Aussage

DIE "AUFRUF-SEMANTIK":



Nachrichtentypen wie Aufruf/Rücksprung von Methoden:

1. Erzeugungsnachricht (Create Message):



2. Synchrone Nachricht (blockiert):



3. Asynchrone Nachricht (blockiert nicht):



BEM: Exceptions werden auch “nur” als normale Nachrichten behandelt.

→ Alle Elemente möglich und Bedeutung wie vorher erklärt

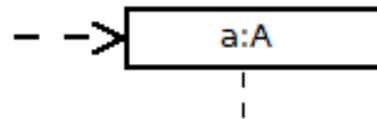
DIE "SIGNAL-SEMANTIK"



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

Übermittlung von Signalen:

- Es wird nur abstrakt von Signalen gesprochen
- Mögliche Nachrichten:
 1. Erzeugungsnachricht (Create Message):



2. Signale:



→ Signale sind als asynchron definiert

- Deshalb wieder der Asynchronpfeil

(Finde ich persönlich etwas ungeschickt, weil für ein Signal vielleicht gar nicht klar ist, ob es asynch oder synch ist)

ZWEI VERSCHIEDENE SEMANTIKEN



Hochschule RheinMain
University of Applied Sciences
Wiesbaden Rüsselsheim



Diese Unterscheid. der Semantiken gibt es nicht offiziell in UML

→ Ableitung von mir aus [UMLglasklar; S. 430]:

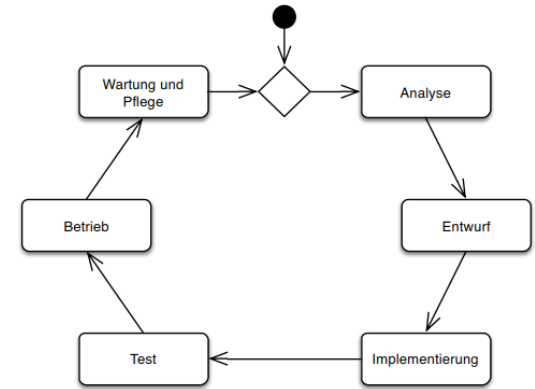
“Im UML-Kommunikationsmodell bilden Nachrichten zwei Formen des Informationsaustausch ab:”

1. “Aufruf / Rücksprung einer Operation” (\triangleq Aufrufsemantik)
2. “Übermittlung von Signalen” (\triangleq Signalsemantik)

“Der Aufruf einer Operation ... Sie können sowohl synchrone als auch asynchrone Aufrufe modellieren. Hingegen ist die Übermittlung eines Signals immer asynchron ...”

- Warum spreche ich jetzt von Aufruf- und Signalsemantik?
 - Für Anfänger griffiger, wenn man es unterscheiden kann
 - Wichtig, weil Sequenzdiagramme in verschiedenen Phasen etwas verschieden verwendet werden.
 - Siehe folgende Folie

WARUM BRAUCHEN WIR ZWEI SEMANTIKEN?



- Die “Aufrufsemantik” ist sehr detailliert
 - Sehr nah an Programmiercode & Programmabläufen
 - Viele Details wie z.B.: wer ruf wen wie genau auf, Rücksprünge müssen definiert sein, ...
 - Schon sehr genaue Festlegungen → nur in Design & Implem.!
- Bei der “Signalsemantik” sind die Signale eher unbestimmt
 - Eignet sich besonders, wenn man noch nicht, weiß ob es wirklich ein Aufruf oder etwas anderes ist
 - Eignet sich also gut in den frühen Phasen (Analyse oder früher Entwurf), wenn man noch nicht festlegen möchte, ob es ein synch. oder asynch. Aufruf (oder was auch immer...) ist
 - siehe dazu auch Dokument mit Bem. zur Robustness Analysis
(→ siehe später auf Homepage zu PiG III – Grobentwurf)



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

04

Kommunikationsdiagramme – Modellelemente im Überblick

Ziel:

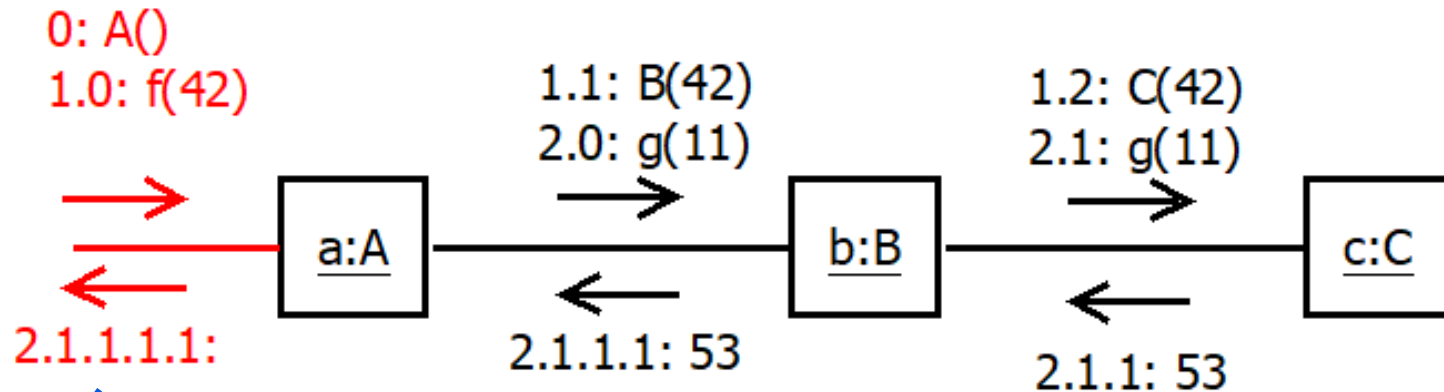
Die Elemente im Überblick erfassen



VORHERIGES BEISPIEL ALS KOMMUNIKATIONSDIAGRAMM:

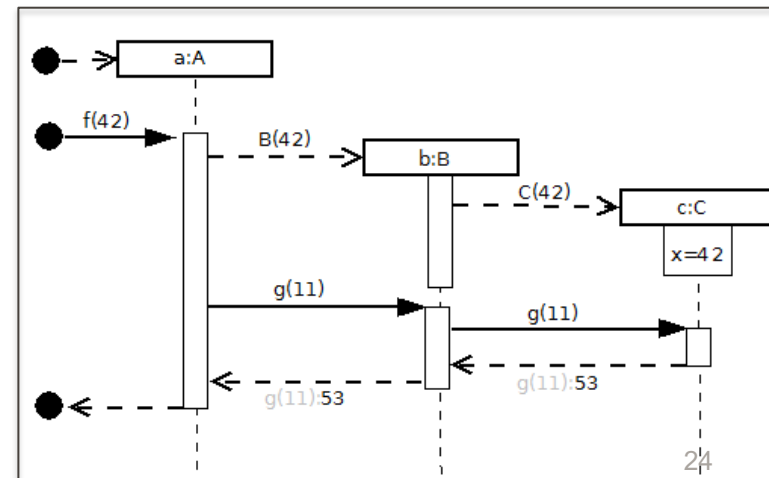


Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim



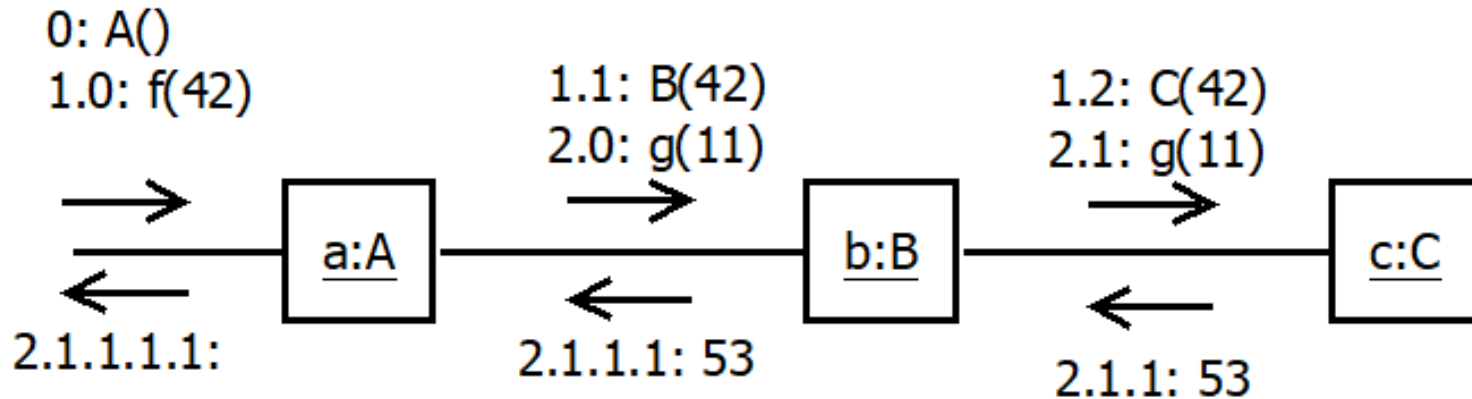
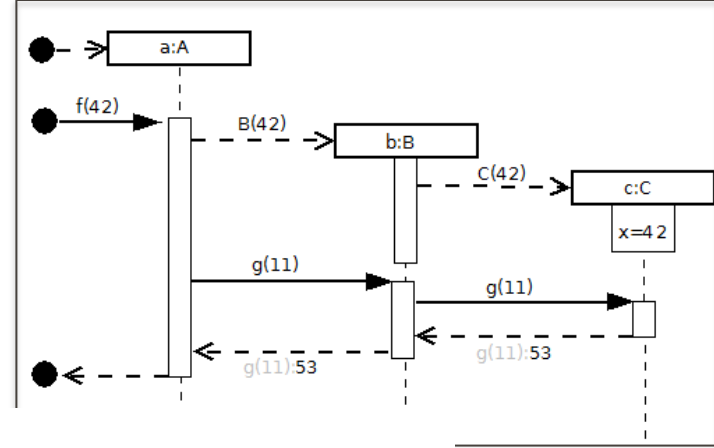
BEM zu rot markiertem Bereich:

Vorschlag von Bodo Iglar für
lost/found Messages
(gibt es nicht in UML2.x bzw. ist dort
nicht geklärt)



KOMMUNIKATIONSDIAGRAMM

WEITERE BEMERKUNGEN



- BEM: Reihenfolge ist nur anhand der Nummerierung ersichtlich → keine Vorgaben wie nummeriert wird!
 - BEM zu Pfeilen:
 - UML 1.x: Gleiche Pfeile wie in Sequenz-Diagramm
 - UML 2.x: Nur → (== Semantikverlust! ☹)
- Für mich: Sie können beides verwenden!

EIGENSCHAFTEN KOMMUNIKATIONS DIAGRAMME



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

- Im Prinzip: Kommunikationsdiagramm hat den gleichen Informationsgehalt wie Sequenzdiagramm
- **ABER:** Nicht alles kann dargestellt werden:
 - Fokus auf (Nachrichten-)Verbindungen zwischen den Kommunikationspartnern
 - Information über Erzeugung und Zerstörung geht verloren
 - Zusammenhänge zw. Nachrichten sind schwerer lesbar
- **Stärke des Kommunikationsdiagramms:**
 - Zwitter aus Struktur- und Interaktionsdiagramm
 - Objekte werden oft in selber Anordnung wie in Klassendiagramm angeordnet (== Struktursicht)
 - Nachrichten zeigen dann die Interaktionen
 - Z.B. gut bei Robustness Analysis (→ PiG – Grobentwurf)!



- Wann am besten welches Diagramm einsetzen?
- Zeitliche Abfolge wichtig → eher Sequenzdiagramm
 - Auch Aktivitäten innerhalb der Objekte teilweise darstellbar
- Eher Struktur wichtig → eher Kommunikationsdiagramm
- BEM: Sequenzdiagramm wird wesentlich häufiger verwendet
 - Finde ich auch meistens besser geeignet



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

05

Weitere Interaktionsdiagramme

Ziel:

Kurze Vorstellung der anderen Interaktionsdiagramme

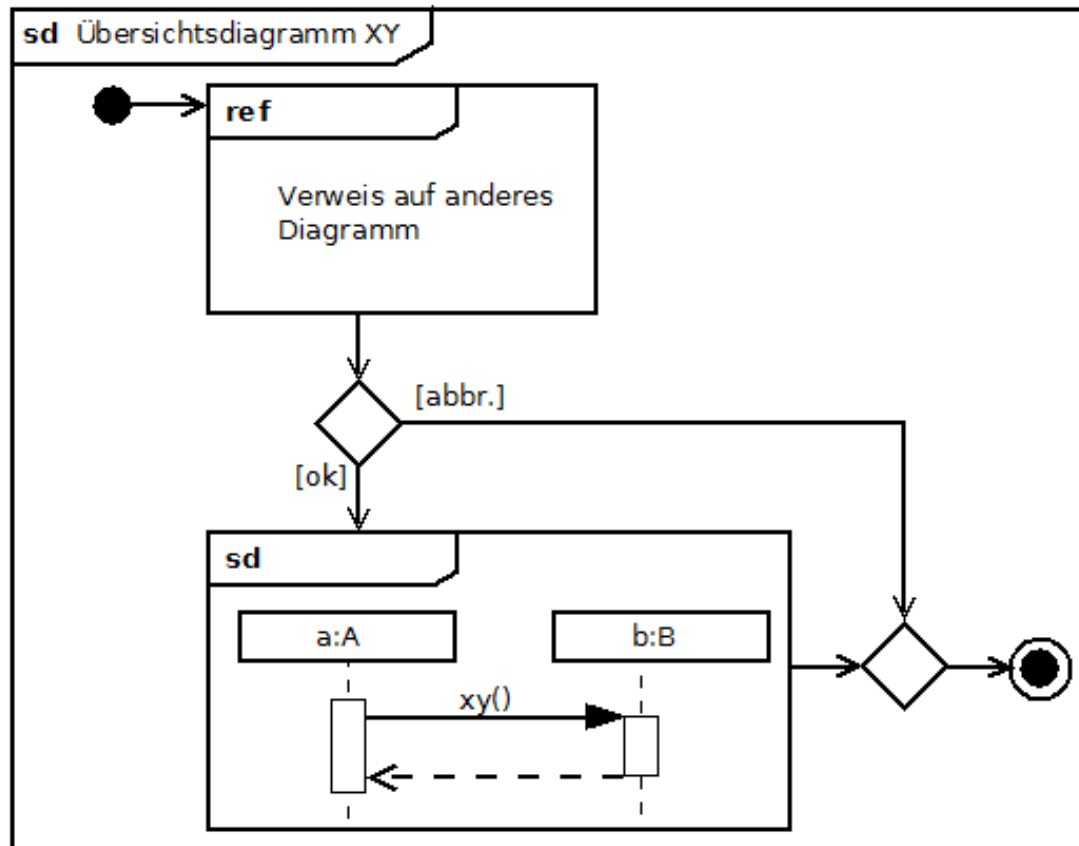


INTERAKTIONSÜBERSICHT-DIAGRAMM



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

Metadiagramm, das die Zusammenhänge zw. verschied. Interaktionsdiagrammen zeigt:

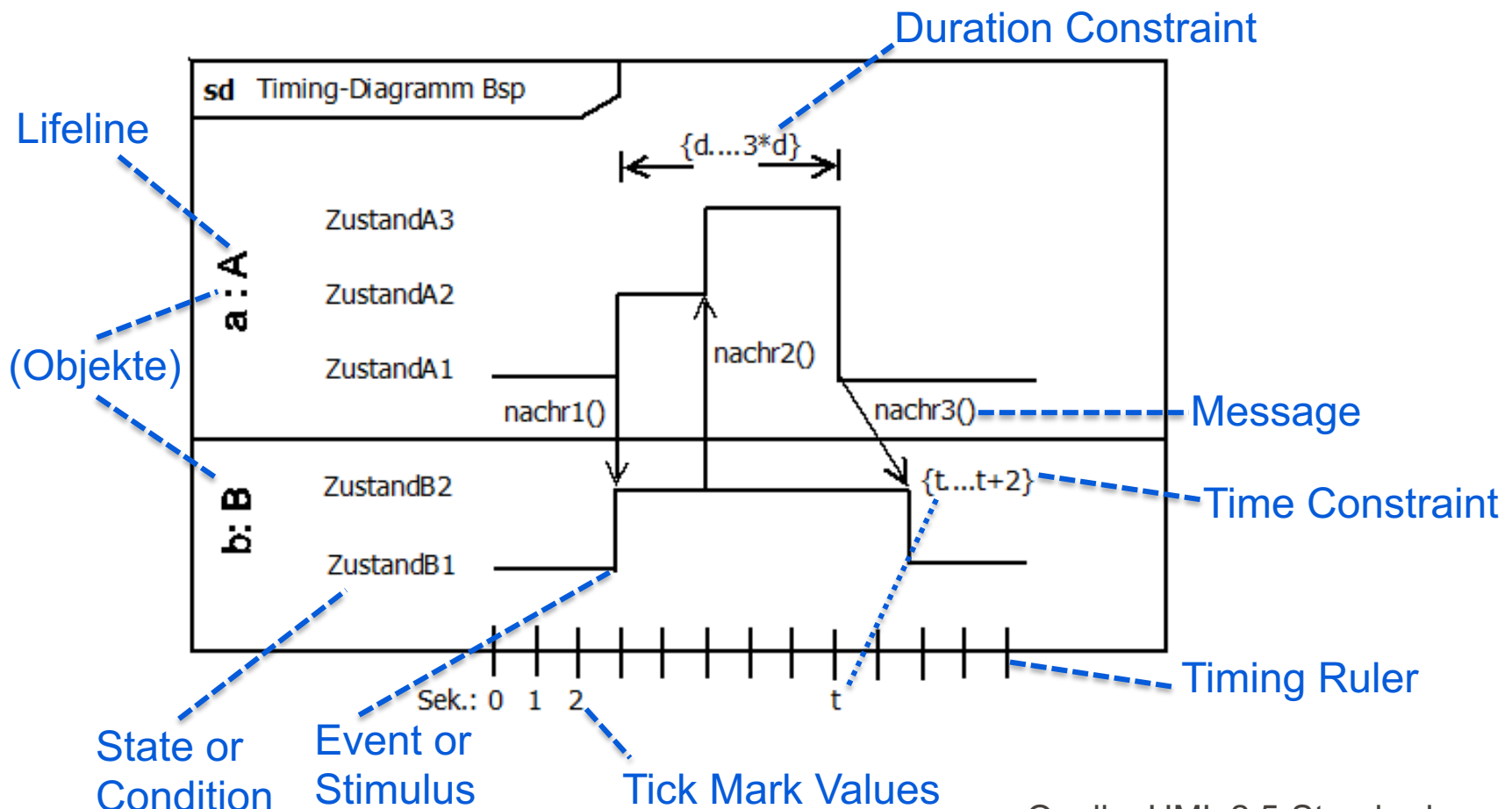


TIMING-DIAGRAMM (ZEITVERLAUFSDIAGRAMM)



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

Zeigt Nachrichtenaustausch und Zustandswechsel verschiedener Objekte zu bestimmten Zeitpunkten an:





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

06 Fazit

Ziel:
Was haben wir damit gewonnen?





WAS HABEN WIR GELERNT?

- Interaktionsdiagramme
 - Sequenzdiagramm (das wichtigste)
 - Kommunikationsdiagramm
 - (Timing-Diagramm)
 - (Interaktionsübersichtsdiagramm)

- **Sequenzdiagramm**
 - Zeigt Objekte und deren Nachrichtenaustausch im Zeitverlauf

- Kommunikationsdiagramm
 - Zeigt Objekte und deren Nachrichtenaustausch in einer Struktursicht (vgl. Objektdiagramm)



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

AUF GEHT'S!!

SELBER MACHEN UND LERNEN!!

