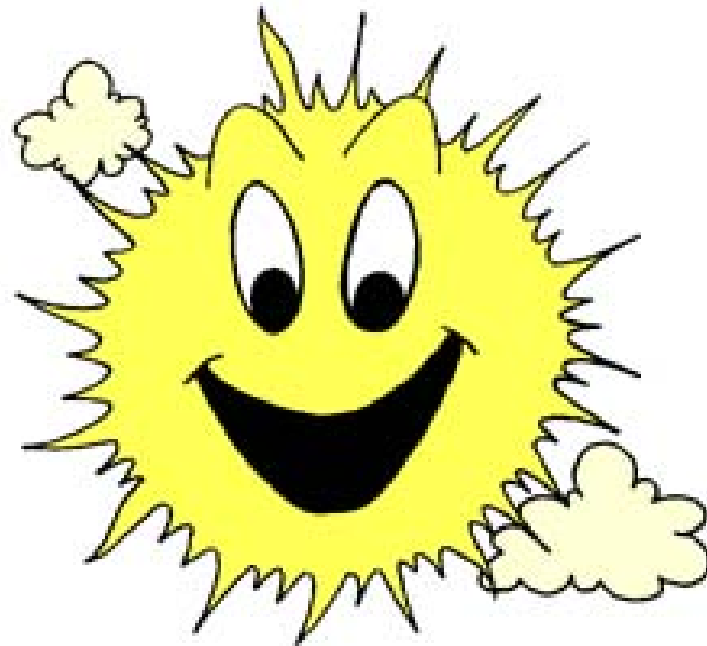


Wiederholung 2. Vorlesung



Häufig gestellte Frage: Parameter in Java

```
public class Test
{
    public static void main( String[] argv ) {
        int argc = argv.length; // Größe eines Feldes (array)
        if ( argc < 1 ) // Prüfen, ob Parameter übergeben wurden
            System.out.println( "Keine Parameter übergeben" );
        else
            System.out.println( argc + " Parameter übergeben: " );

        for ( int i = 0; i < argc; i++ ) { // Parameter anzeigen
            System.out.println( (i+1) + ": " + argv[i] );
        }
        System.out.println(); // Leerzeile
        // Versuchen, Parameter in int zu parsen
        for ( int i = 0; i < argc; i++ ) {
            try { // try-Block
                int zahl = Integer.parseInt( argv[i] );
                System.out.println( "Argument " + (i+1) + ": int = " + zahl );
            }
            catch ( NumberFormatException e ) { // "fängt" Exception auf
                System.out.println( "Fehler: " + e.getMessage() );
            }
        } // end for
    } // end main
} // end class
```

1.3.3 Entwickler und Anwender ... (I)

... spielen einen neuen Dialog durch:

Entwickler: ... So, jetzt hast Du einen Teil der Aufgabe bereits ausgeführt. Willst Du nun noch etwas verändern?

Anwender: Ja, ich möchte eine Komponente löschen. Was muss ich tun?

E: Deute auf das Menu mit dem Namen 'CD'.

A: CD ?

E: CD bedeutet 'Component Delete'.

A: OK, ich habe verstanden ... Oh, was passiert jetzt?

1.3.3 Entwickler und Anwender ... (II)

E: Du bist jetzt im Analyse-Modus.

Du musst aus Versehen 'AM' ausgewählt haben.

A: Ach so, ich dachte, ich hätte 'CD' ausgewählt.

Wie verlasse ich den Analyse-Modus?

E: Kein Problem. Gib einfach 'control Q' ein.

A: [tippt C O N T R O L ...]

E: Nein! Drücke die 'Strg'-Taste und die 'Q'-Taste.

A: Wie dumm von mir. ... Ich versuch's noch einmal.

... Es passiert gar nichts. Was habe ich falsch gemacht?

1.3.3 Entwickler und Anwender ... (III)

E: Du hast nichts falsch gemacht. Die Komponente ist gelöscht aber noch nicht vom Bildschirm entfernt. Gib noch 'control-J' ein.

A: Ich habe verstanden. Wenn ich die nächste Komponente lösche, gebe ich wieder 'control-Q' ein.

... Hoppla, warum ist der Bildschirm jetzt weiß?

E: Du hast nur 'Q' eingegeben. 'Q' bedeutet Quit und beendet das Programm. Jetzt musst Du leider alles noch einmal eingeben!

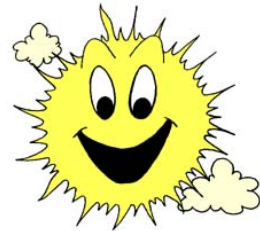
A: Wenn das so ist, machen wir nächste Woche weiter.

1.3.3 Entwickler und Anwender ... (IV)

Folgende Fehler wurden gemacht:

- Kommandos (CD, AM) sind nicht intuitiv
 - ➔ erzeugt unnötige Belastung für den Anwender
- Auf welches Objekt wird das Kommando angewendet?
- Keine Statusinformation zu Zustandsänderungen
- Kryptische Kommandos (control-J, control-Q) sind ungeeignet für unerfahrene, gelegentliche Benutzer
- Kein Feedback bei Delete
 - ➔ erzeugt Frustration und Ablehnung beim Anwender
 - ➔ ergibt Negativausstrahlung auf das Gesamtsystem

Ende der Wiederholung



1.3.4 Ziele bei der Entwicklung guter Anwendungen



2. Entwicklung von Anwendungen (1)

- Prinzipielle Ziele beim Entwurf von Anwendungen
- Benutzerzentriertes Requirement Engineering
 - Use Case
 - Sequenz-Diagramm
 - (Rapid) Prototyping
 - Gestaltungskriterien

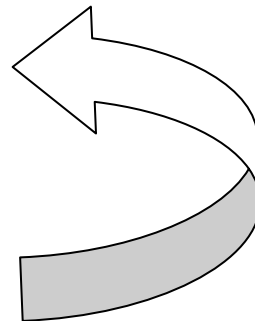
2. Entwicklung von Anwendungen (2)

Auch für das **Software Design** ist UML ein guter Ansatz:

- Use Case
- Klassen-Diagramm
- Sequenz-Diagramm
- Kollaborationsdiagramm
- Status-Diagramm

Entwicklung schrittweise vornehmen!

- schreiben
- testen
- erweitern
- testen



zyklisch
und
iterativ

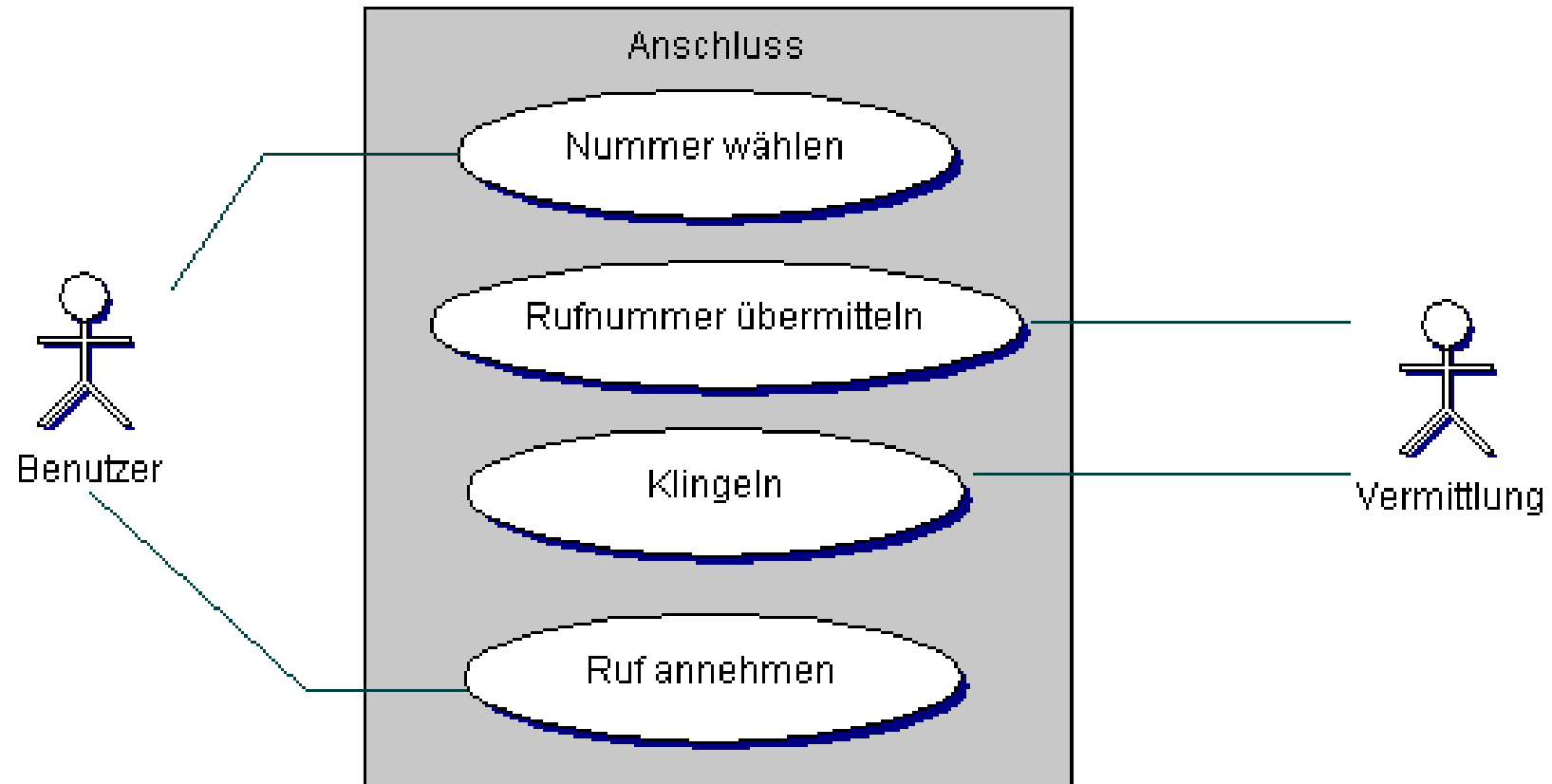
2. Beispiel für SW-Design: Telefonie

Im folgenden Beispiel modellieren wir in groben Zügen die SW-Funktionen eines Telefon-Netzes:

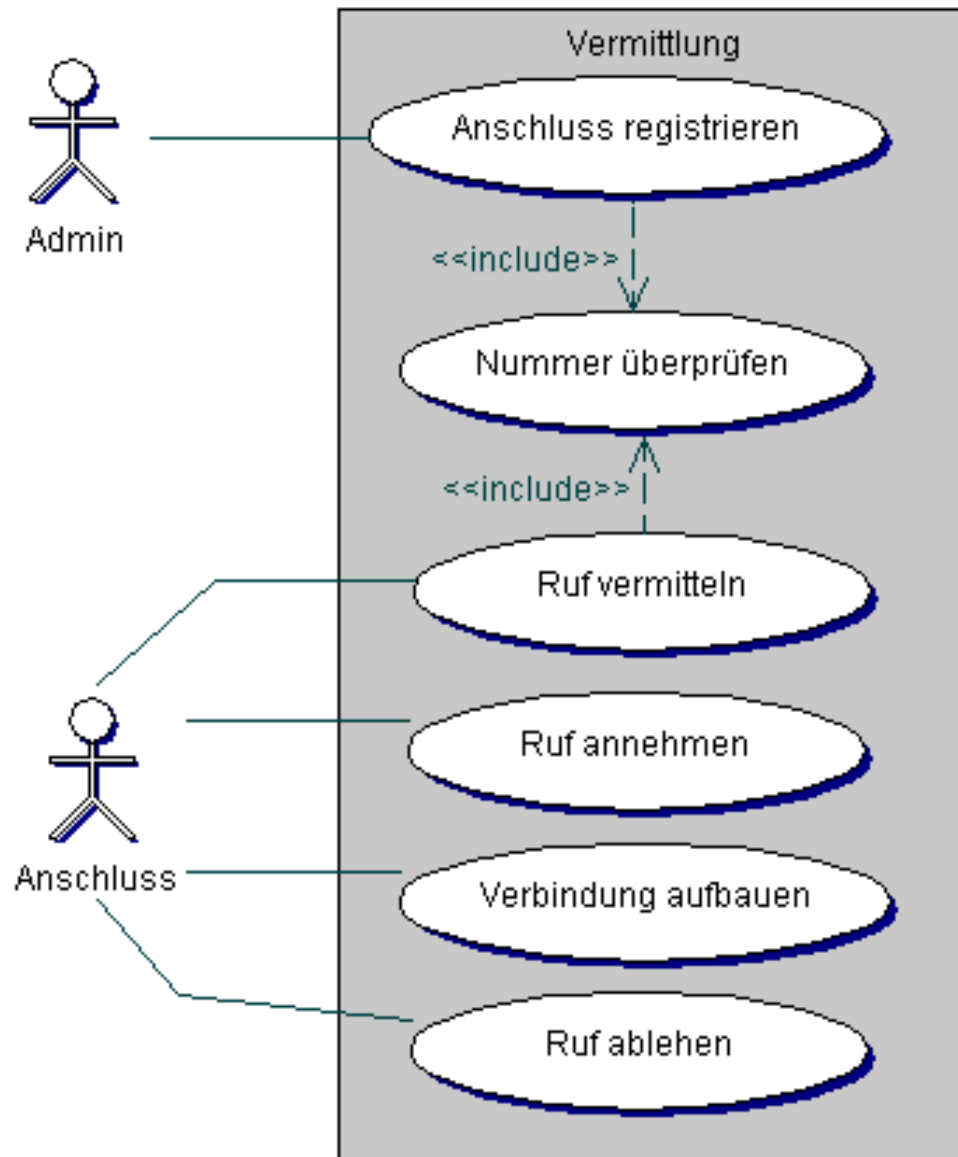
Es gibt Anschlüsse, ein Netz (Vermittlung genannt) und Benutzer

Wir beginnen mit den Anwendungsfällen ...

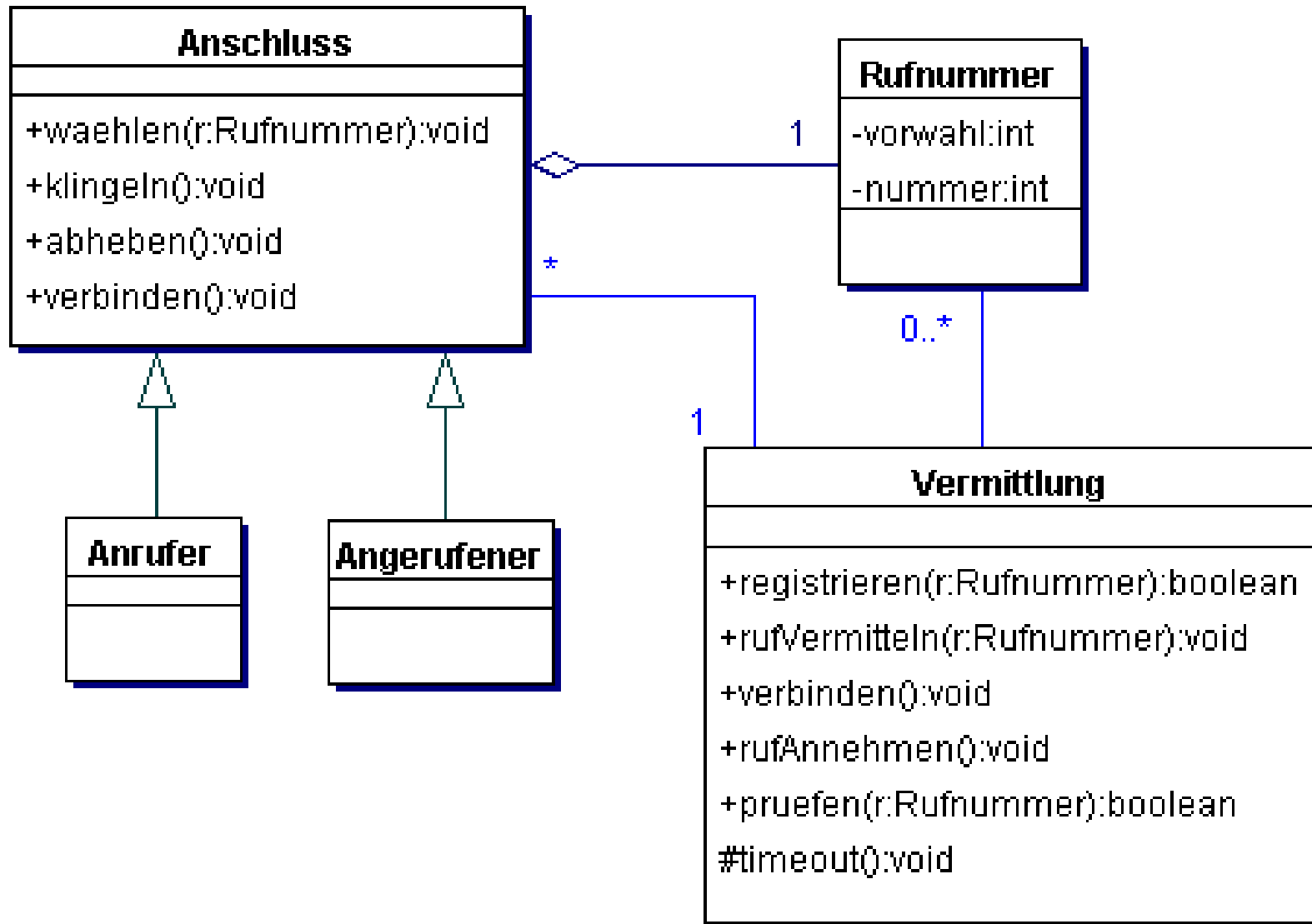
2.1 Beispiel Anwendungsfall (Use Case 1)



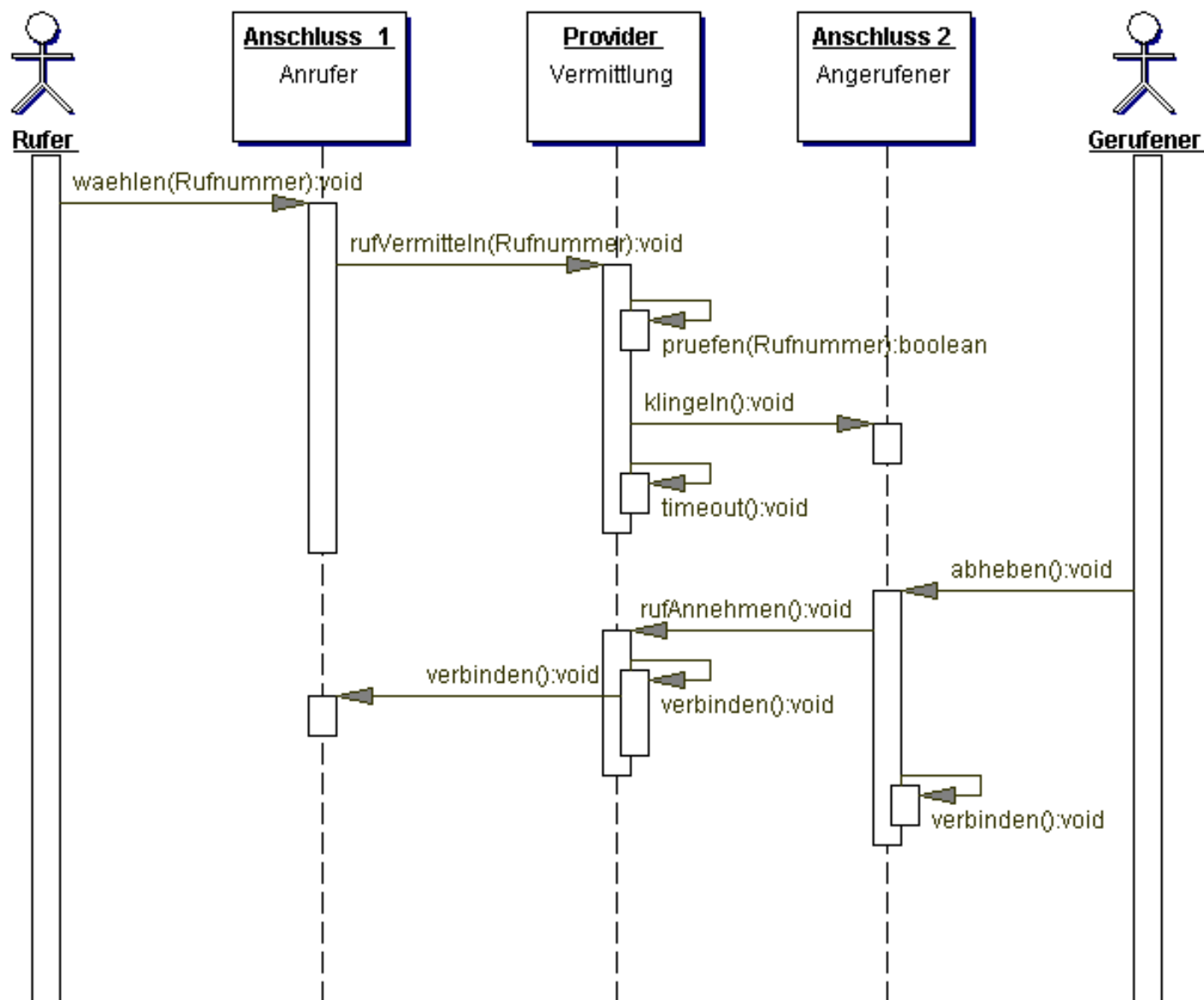
2.1 Beispiel Anwendungsfall (Use Case 2)



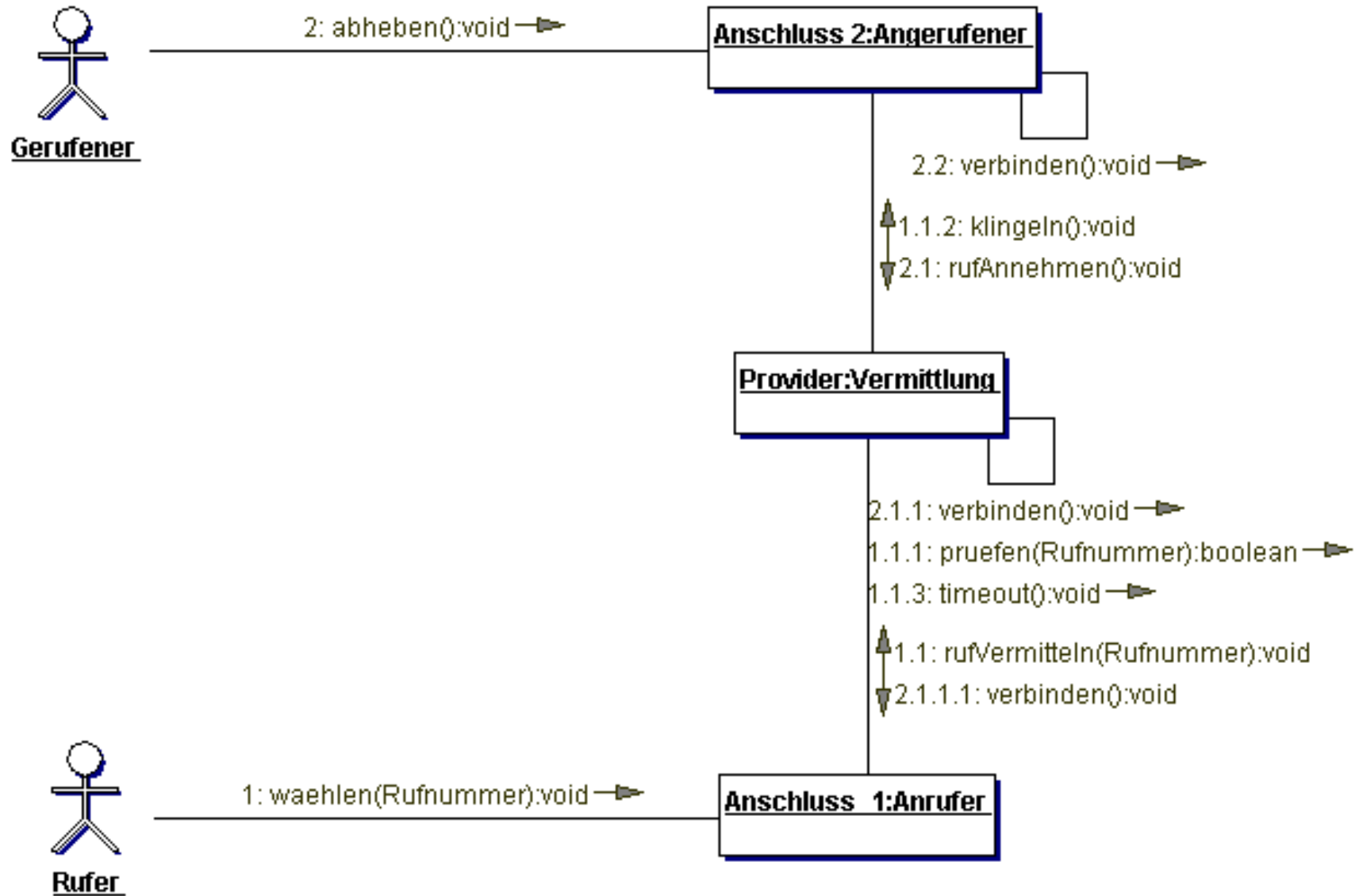
2.1 Beispiel Klassendiagramm



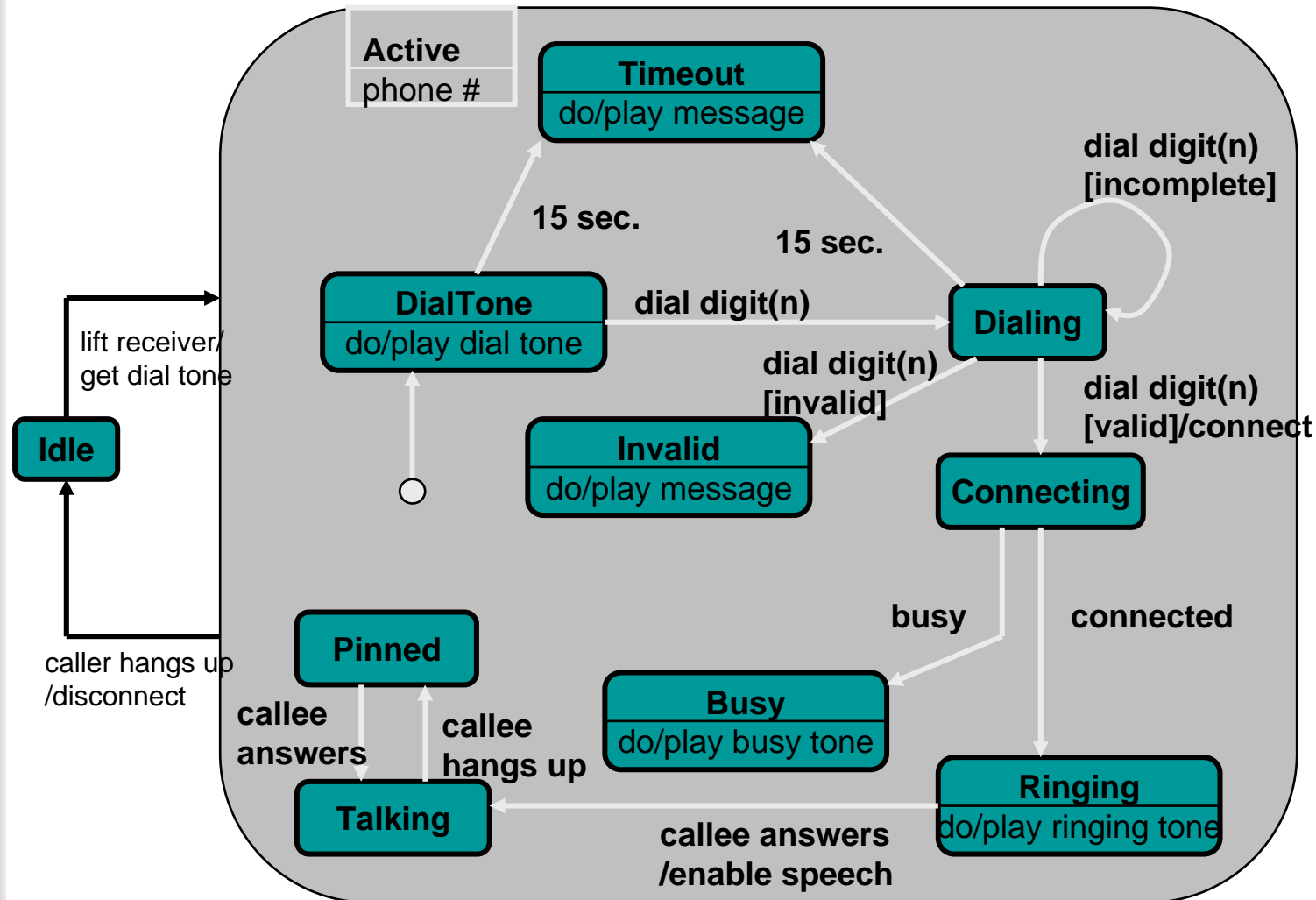
2.1 Beispiel Sequenzdiagramm (Abläufe)



2.1 Beispiel Kollaborationsdiagramm



2.1 Beispiel Zustandsdiagramm



Im Zustandsdiagramm werden die Wirkungen von Botschaften in Abhängigkeit von den inneren Zuständen des Objektes und die resultierenden Zustandswechsel dargestellt.

2.1 Analyse und Design

... wird später noch weiter vertieft

- Vorlesung Software-Technik
- Entwurf und Realisierung grafischer Oberflächen:
 - Vorlesung
 - Praktikum

2.2 Ziele bei der Entwicklung guter Anwendungen



Gruppenarbeit:

Erarbeiten und definieren Sie Ziele einer guten Computeranwendung

- interne Eigenschaften des Systems
- **Schnittstelle Mensch – Maschine**

2.2 eine gute Anwendung hat/ist/bietet ... (Gruppenarbeit)

- intuitive GUI und Bedienung
- eindeutige/verständliche Dialoge/Befehle
- kontextsensitive Hilfe
- Ergonomie
 - keine unnötige Belastung
 - logischer Aufbau – Workflow
 - Schrift und Farbgebung
 - Shortcuts zu Befehlen
 - Macro/Batch-Fähigkeiten
 - Barrierefreiheit
 - zielgruppenorientiert
- Feedback über
 - Zustände / Status
 - Aktionen
- Fehlertoleranz
- einheitliches Look & Feel
- Internationalisierung / Lokalisierung
- Fehlerfreiheit
- Effizienz / Performanz
- Robustheit / Stabilität
- Sicherheit
- wartbar / erweiterbar
- modular
- portabel

3. Entwicklungsumgebungen für Java

- Es gibt unterschiedliche Entwicklungsumgebungen für Java, z.B. Eclipse (<http://www.eclipse.org>)
- Vorteile beim Entwurf von grafischen Oberflächen (Code-Generierung)
- Wir benutzen Borland JBuilder Foundation 5.0
<http://www.borland.com/jbuilder/foundation/index.html>

4. Grafische Grundkomponenten

- Label
- Text-Field (Textfeld)
- Text-Area (Textbereich)
- Button (Schaltfläche)
 - Simple Button
 - Toggle Button
 - Checkbox
 - Radio Button
- Auswahlboxen
 - Combo Box
 - List Box
 - Spin Button
 - Tabellen
- Slider (Schieberegler)
- Progress Bar
- Menu
- Toolbar
- Tooltip
- Dialogboxen
 - Information
 - Eingabe
- ...