

1 Einführung

**1.1 Unterscheidung Anwender ↔ Benutzer**  
**Anwender** sind Entitäten, welche direkt oder auch indirekt ein Hilfsmittel zur Erzielung eines Vorteils verwenden  
**Benutzer** sind natürliche Personen, die direkt mit einem Hilfsmittel arbeiten, um einen Vorteil zu erzielen

*Beispiel:* Ein Direktor (Anwender) lässt sich von seinem Chauffeur (Benutzer) mit einem Auto (Hilfsmittel) zu seinem nächsten Termin bringen

1.2 Arten der Kontrolle

- **Interne Kontrolle:** Assistenten
- **Externe Kontrolle:** WIMP

1.3 Definition Interaktives System

Ein **interaktives System** gibt dem Benutzer die Möglichkeit, mittels einer Benutzerschnittstelle die laufende Bearbeitung einer Aufgabe durch ein technisches System zu beeinflussen.

1.4 Formen von Interaktion

Nach *Preece et al.*

- **Anweisung**
  - Eingabe von Kommandos
- **Dialog**
  - Ablauf von Eingabe und Ausgabe
- **Manipulation (von Kommandos, Inhalten)**
  - Auswählen („Choice“)
  - Ansteuern („Target Acquisition“)
  - Verändern
- **Erkunden (einer Menge von Kommandos [Menu], Inhalten)**
  - Suchanfragen
  - Browsing

1.5 Arten von Benutzerschnittstellen

- **Text User Interface**
- **Tangible User Interface** (→ Post-WIMP)
- **Graphical User Interface**
- **Voice User Interface**
- **Haptic User Interface**
- **Natural User Interface** (möglichst wenig zu erlernende Eingabegeräte)
- **Intuitive User Interface** (möglichst leicht erlernbare Eingabegeräte)
- **Brain Computer Interface**
- **Command Line Interpreter**

1.6 Fallstudien

**Sketchpad**

Programm entwickelt von Ivan Sutherland im Rahmen einer Doktorarbeit.  
Mittels Lichtgriffel: Zeichnen und Deformieren geometrischer Objekte; Vektorgraphiken; Erste objektorientierte Ansätze zur Bedienung

**Xerox Alto**

Kommandozeileninterpreter Alto Executive: Graphische Oberfläche; Mehrere Applikationen; Mehrere Fenster

**Apple Macintosh**

Features u.a. Desktop-Metapher, Drop-Down-Menüs, Ordner, Mülleimer sowie überlappende Fenster

**Windows 1.0**

Zeigegerät, Menüs, Statuszeile, Symbole, Zwischenablage, Taskleiste. Wegen Vektorgrafik-Fähigkeit bereits als GUI (und nicht als TUI) klassifiziert.

2 Zeigegeräte

2.1 Einteilung von Interaktionstechnologien

Nach *Krauß*

- **Koordinatengebend**
  - Dimensionalität
    - eindimensional, zweidimensional, mehrdimensional
  - Verhältnis Lage zu Wirkort
    - Indirekt wirkende vs. direkt wirkende
- **Nicht koordinatengebend**
  - Tasten
  - Sprache
  - Gesten

2.2 Koordinatengebende

2.2.1 Direkte Zeigegeräte

- Es wird unmittelbar auf die Ausgabe gezeigt
- **Merkmale**
  - Lernaufwand Bedienung Zeigege-räts minimal
  - Verdeckung der Ausgabe durch Zeigegerät
  - Ermüdungserscheinungen
- *Beispiel:* Touchscreen

2.2.2 Indirekte Zeigegeräte

- Nicht im direkten Kontakt mit der Ausgabe
- Bewegung an anderer Stelle und Transformation der Koordinaten an den Bildschirm
- **Merkmale**
  - Lernaufwand Bedienung des Zeigege-räts z.B. Hand-Augen-Koordination bei Comput-ermaus
  - Kopplung zwischen Zeigegerät und Ausgabe als modulierbarer Parameter
- *Beispiel:* Computermaus

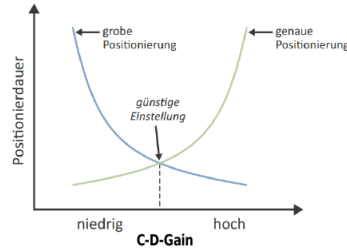
2.2.3 Positionierung

**Absolute Positionierung** Eindeutige Zuordnung der Position des Zeigege-rätes zur Position auf dem Bildschirm. Der Bezug zum Ursprung kann gegeben sein.

**Relative Positionierung** Es wird von aktuell gespeicherter Position ausgegangen und die Veränderung der Koordinaten umgesetzt.

**Control-Display-Gain** «**MacKenzie**» Der durch die Bewegung eines indirekten Zeigege-rätes ("Con-

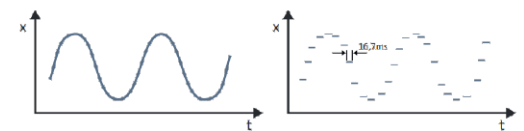
trol") resultierende Effekt („Gain“) am Bildschirm. Nach MacKenzie balanciert eine günstige Einstellung des Gain Schnelligkeit und Präzision der Positionierung.



- *Gain niedrig:* Große Bewegung Controller bewirkt durchschnittlichen Effekt; Vorteilhaft für Feinpositionierung
- *Gain hoch:* Kleine Bewegung Controller bewirkt durchschnittlichen Effekt; Vorteilhaft für Grobpositionierung

2.2.4 Physikalische Charakterisierung Abtastrate

- Erfassung der Position und Tasten des Zeigege-rätes zu diskreten Zeitabständen
- Definiert die zur Positionsbestimmung verfügbare Zeit



**Verzögerung**

- Zeitabstand zwischen Abtastung und Änderung am Bildschirm
- Verzögerung sollte geringer als die Hälfte der Abtastrate sein
- Schon geringe Verzögerungen bringen höhere Fehlerrate bei Selektionen mit sich

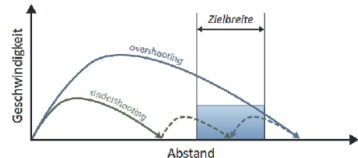
**Genauigkeit**

- Genauigkeit: Abweichung der durch einen Sensor gemessenen Position von der tatsächlichen Position
- Gemeinsame Optimierung von Genauigkeit und Abtastrate nicht möglich
  - Obergrenze bildet die räumliche Auflösung des Sensors
  - Hohe Abtastrate geht zu Lasten der Genauigkeit

Abhängigkeit von der individuellen Anwendung

- Bei Interaktion in komplexen virtuellen Welten in modernen Computerspielen oder 3D-Modellierungstools können hohe Anforderungen an Zeigegeräte gestellt werden

2.2.5 Zeige Aufgaben



Das Experiment nach Fitts: Wie lange benötigt ein Mensch *MT*, um mit einem Stift ausgehend von einem Startpunkt ein Ziel zu erreichen? Das Experiment steht dabei in der Abhängigkeit von der Distanz *D* des Ziels und der Breite *W*.

**Fitts**

$$MT = a + b \cdot ID$$

$$ID = \log_2 \left( \frac{2D}{W} \right)$$

**MacKenzie**

Verbesserung der Formel von Fitts

$$ID = \log_2 \left( \frac{D}{W} + 1 \right)$$

**Accot & Zhai**

Berücksichtigt auch Höhe und y-Koordinate des Ziels

$$MT = a + b \cdot \log_2 \left( \sqrt{\left( \frac{D}{W} \right)^2 + \eta \left( \frac{D}{H} \right)^2} + 1 \right)$$

2.2.6 Ergebnisse des Praktikumsversuchs

MacKenzie > Accot/Zhai > Fitts

- In Fitts' Experiment wird ein direktes Zeigegerät benutzt
- *Precuing* macht eine Zeigeaufgabe immer einfacher
- *Reset* macht eine Zeigeaufgabe zunächst schwerer, jedoch kann es schnell erlernt werden

2.2.7 Unterstützung von Zeigeaufgaben

**Anpassung Bedienelemente** Eine Anpassung der Größe, Anordnung, Form.  
Verkleinern ist nach Fitts schwieriger und die Unterscheidbarkeit nimmt ab.

**Temporäre Modifikation des Interface** Eine Anpassung des User Interface.  
Nach Parker kann man das Ziel oder den Cursor vergrößern, sowie den Cursor zum Ziel bewegen oder auch das Ziel zum Cursor bewegen.  
Nachteile: Eine Veränderung des User Interface kann den Benutzer irritieren; Aussagen über eine Selektion-sinteraktion sind schwer

**Mehrere Mauszeiger** Nachteile: Potentiell mehrdeutige Zielauswahl

**2.2.8 Direct Manipulation**  
Direkt manipulative Interfaces zeichnen sich aus nach *Hutchins et al.* durch:

- Geringe Distanz
  - von Benutzerzielen
  - zu deren Erreichung nötigen Interaktionen
- Hohes Engagement
  - direkte Auseinandersetzung des Benutzers
  - mit dem Objekt von Interesse
- Beispiel:
  - Skript schreiben: Distanz hoch, Engagement niedrig
  - Drag & Drop: Distanz niedrig, Engagement hoch

**Prinzipien nach Shneiderman**

- Kontinuierliche Repräsentation des Objekts von Interesse
- Eingabe auf Basis einfacher physikalische Aktionen (z.B. Bewegung der Maus) anstelle sprachbasierter Kommandos
- Schnelle, schrittweise, umkehrbare Operationen, deren Auswirkung auf das Objekt von Interesse unmittelbar erkennbar wird
- Schrittweises Erlernen der Interaktion ausgehend von minimalem Vorwissen

**Vorteile nach Shneiderman**

- Anfänger können die grundlegende Funktionen schnell erlernen, beispielsweise nach Vorführung durch einen erfahreneren Benutzer
- Interaktionskonzepte werden nach längerer Nichtanwendung leicht erinnert
- Benutzer können unmittelbar erkennen, ob sie sich durch Aktionen ihren Zielen annähern – und gegebenenfalls ihre Aktionen anders ausrichten
- Benutzer sind weniger zögerlich im Umgang mit dem System, da Aktionen leicht verständlich und umkehrbar sind
- Benutzer erleben Kontrolle über das System und können dessen Reaktionen direkt beobachten

**2.3 Nicht koordinatengebend**  
**2.3.1 Sprachbasierte Interaktionsformen**  
Nach *Preim & Dachselt*

- Kommandosprachen

- Textuelle Suche
- Natürlich-sprachliche Systeme

**Phoneme & Prosodie**  
Phonem: “Phoneme sind die kleinsten bedeutungsunterscheidenden Einheiten einer Sprache.”  
Der Austausch eines Phonems ändert die Bedeutung eines Wortes. Prosodie: "Prosodie ist die Gesamtheit derjenigen lautlichen Eigenschaften der Sprache, die nicht [...] ans Phonem als minimales Segment, sondern an umfassendere lautliche Einheiten gebunden sind"  
Dient u.a. der Kennzeichnung des Satztyps (z.B. Frage, Aussage), Gewichtung oder Gliederung.

**Spracherkennung** Audiosignal → Text

- Vorverarbeitung (Filterung um Störgeräusche zu extrahieren, Frequenzspektrum berechnen, Sprachrelevante Merkmale extrahieren)
- Extraktion von Phonemen
- Abbildung von Phonemfolgen auf Wörter

**Semantische Analyse** Bedeutung der Aussage auf struktureller Ebene (Agenten, Handlungen, Intentionen, Objekte), Ergebnis unter Umständen nicht eindeutig

**Pragmatische Analyse** Bedeutung der Aussage im Kontext, z.B. Ort, Zeit, Fähigkeiten, Intention; Hinzuziehen weiterer Sprachmerkmale (Prosodie); auch hier Ergebnis unter Umständen nicht eindeutig

**Sprachsynthese** Text → Audiosignal

- NLP (Natural Language Processing) transkribiert Text anhand eines Phonem-Alphabets und einer Prosodiebeschreibung
- DSP (Digital Speech Processing) synthetisiert aus der NLP-Ausgabe das Audio-Signal gesprochener Sprache

Ansätze für die Umsetzung:

- Modellbasierte Synthese: erzeugt ein Audio-Signal auf Basis eines Modells
- Konkatenative Synthese: verknüpft Ausschnitte aus Aufzeichnungen gesprochener Sprache um zu einem neuen Audiosignal

**2.3.2 Tastatur**  
Die Effizienz hängt ab von ...

- ... Größe, Form und Position der Tasten
- ... Art der Eingabe (Symbolhäufigkeiten und Symbolfolgen), im Fall natürlicher Sprache also von der Sprache, in der der Text verfasst wurde
- ... Technologischen Randbedingungen

Die **QWERTY** Tastenbelegung orientierte sich danach das sich in der Englischen Sprache die Typenhebel einer Schreibmaschine nicht kollidierten.

**Virtuelle Tastaturen**

- Platzersparnis: z.B. Temporäres Einblenden bei Touch-Interaktion genau dann, wenn alphanumerische Eingaben benötigt werden
- Konfigurierbarkeit: z.B. Tastaturen, deren Layout sich anpasst

Probleme: Fat Finger Problem, Fehlender Druckpunkt (Tasten können versehentlich ausgelöst werden)

**3 Graphische Dialogsysteme**  
**3.1 WIMP-Prinzipien**  
**WIMP:** Windows Icons Menus Pointer

**Metaphors** Interaktionsmechanismen werden auf bekannte Arbeitsabläufe abgebildet

**Direct Manipulation** Der Benutzer soll direkt mit Objekten der Benutzerschnittstelle interagieren können

**See and Point** Die Interaktion mit der Maschine soll auf den gerade sichtbaren Objekten basieren

**Consistency** Objekte mit ähnlicher Funktion sollen ähnlich aussehen und sich ähnlich verhalten

**User Control** Nur der Benutzer soll Aktionen der Maschine initiieren und deren Ablauf kontrollieren

**Feedback and Dialog** Alle ausgeführten Aktionen des Benutzers und deren Effekte sollen ohne Verzögerung erkennbar sein

**Forgiveness** Aktionen des Benutzers sollen rückgängig gemacht werden können, bei kritischen Aktionen soll gewarnt werden

**Aesthetic Integrity** Das Erscheinungsbild der Benutzeroberfläche soll einfach, aufgeräumt und konsistent sein

**Modelessness** Der Benutzer soll alle möglichen Aktionen zu jedem Zeitpunkt ausführen können. (Aufwendig zu implementieren)

**WYSIWYG** Dokumente sollen bei der Erstellung so angezeigt werden, wie sie als Ausdruck ausgegeben werden. (**Nur Drucker!**)

- 3.2 Fenster**
- Rechteckige Bereiche, die Interaktionen von Benutzern verarbeiten können
  - Präsentation und Verhalten angelehnt an Metapher „Papier“
    - stapelbar, verschiebbar
    - Passend zur Metapher „Schreibtisch“
    - Historisch in Büroarbeiten begründet
  - Begriff verdeutlicht die Funktion einer Sicht („View“) auf Objekte einer Anwendung

**3.2.1 Varianten**  
**Applikationsfenster** Beinhaltet (logisch) alle Bedienelemente einer Anwendung

**Dialogfenster** Informieren oder Aufforderung zur Eingabe  
**Elementares Fenster (Widget)** Diejenigen Bedienelemente, die eine Einheit aus Erscheinung und zugehörigem Verhalten bilden

- 3.2.2 Verwaltung von Fenstern**
- Window Manager sind entweder im Fenstersystem integriert, oder können auch austauschbar sein
  - Realisieren das Verschieben, Skalieren, Öffnen und Schließen der Fenster
  - Steuern das Aussehen der Fenster, d.h. die Gestaltung der Titelzeile und der Scrollbars
  - Realisieren Fensterplatzierungsstrategien
  - Unterstützen Navigation in Fenstern

- 3.2.3 Navigation in Bildschirmfenstern**
- Die physikalische Größe des Ausgabegeräts beschränkt den sichtbaren Bereich eines Fensters
  - Verkleinerung des Weltmodells im Zuge der Window-to-Viewport-Transformation nur begrenzt zielführend aufgrund:
    - Grenzen der Auflösung des Ausgabegeräts
    - Grenzen der menschlichen Wahrnehmung
  - Dem Benutzer wird daher ermöglicht, das im Weltmodell definierte Fenster zu manipulieren

**Scrolling** Indirekte Manipulation der Position im Weltmodell; Unterstützt durch Controls des Fensters (Scrollbar) und ggf. Hardware (Tastatur, Mousrad)

**Panning** Direkte Manipulation der Position im Weltmodell

**Zooming** Skalierung der Größe des Fensters im Weltmodell; U.U. verbunden mit einer Modifikation von Datenselektion und Präsentation (vgl. Bederson & Hollan, 1994); B. Bederson. PA3D: Early zoomable user interface (ZUI)

**Mehraufwand durch Fenstermanagement**  
Nach *Bury & Darnell*:  
Primäre Aufgabe der Probanden (Textverarbeitung) erfordert Einsatz von Hilfsfunktionen:

1. Freie Verwaltung der Fenster der Hilfswerkzeuge
2. Hilfswerkzeuge immer Vollbild

Beobachtung: Freie Verwaltung verlangsamt das Arbeiten, führt aber zu einem akkurateren Arbeiten.

**3.2.4 Virtuelle Desktops**  
Ziel: Verwaltungsaufwand reduzieren unter Beibehaltung einer akkuraten Arbeitsweise. Virtuelle Desktops realisieren eine Raum-Metapher (Henderson & Card, 1986): Fenster (Anwendungen) werden in Gruppen gebündelt, welche einen "Raum" nachbilden. Inhaltliche Logik der Bündelung vorgegeben durch System oder individuell durch Benutzer.

**3.2.5 Koordinatensysteme**  
Das Koordinatensystem eines Fenstersystems bestimmt Positionierung und Größe präsentierter Information

**Geräteabhängige Koordinaten** Die Koordinaten auf einem gegebenen Ausgabegerät

**Geräteunabhängige Koordinaten** Die Entkopplung der Präsentation von geräteabhängigen Parametern (insbesondere: Auflösung) der Ausgabegeräte

*Beispiel:* Eine 1 cm Linie soll auf einem Bildschirm mit 150 dpi genauso lang ausgegeben werden wie auf einem Drucker mit 2400 dpi

**Weltkoordinaten** Entkopplung der Präsentation von der Repräsentation der zugrundeliegenden Daten

*Beispiel:* Sowohl Nanometer für Moleküle wie auch astronomische Koordinaten sollen vom Modell abgebildet werden können

3.3 Piktogramme

- Symbole, die Komponenten eines Fenstersystems graphisch repräsentieren
- Orientierung an bildhaften Darstellungen in anderen Bereichen der natürlichen Umgebung
  - Ziel: Ausnutzen der ausgeprägten Stärke des Menschen, erlernte Muster (wieder) zu erkennen
- Unabhängigkeit von einer bestimmten Sprache
  - Können damit zur Internationalisierung beitragen

3.3.1 Varianten

**Repräsentative Piktogramme** Stellen einen abstrakten Sachverhalt bzw. ein Konzept dar  
*Beispiel:* Symbole für Mülleimer, Disketten, Drucker

**Abstrakte Piktogramme** Stellen einen abstrakten Sachverhalt bzw. ein Konzept dar und müssen somit erst erlernt werden

*Beispiel:* Symbole für Pfeile, Wiederholung, Rückgängig

**Hybride Formen** Stellen eine Mischform da

3.3.2 Richtlinien

- Einfachheit und Klarheit
- Verständlichkeit
- Einprägsamkeit
- Platzsparende Gestaltung
- Klarer Kontrast zu Hintergrund
- Hohe Unterscheidbarkeit vs. Konsistenz
- Selektierter Zustand sollte klar ersichtlich sein

3.4 Menüs

- Basieren auf Hierarchien und Taxonomien
- Dienen dem Auslösen atomarer Aktionen
- Komplexere Interaktionen wie die Eingabe mehrerer Werte etc. durch die Verbindung mit Dialogen

Motivation

- Es soll unter einer endlichen Menge von Aktionen gewählt werden
- Man soll die Menge der Aktionen nicht erlernen müssen
- Man soll im Vermeiden syntaktischer Fehler unterstützt werden

3.4.1 Hick & Hyman

Versuchsaufbau „choice-reaction time“

- 2-10 Lampen angeordnet in einem irregulären Kreis
- Jeder Lampe ist eine Taste zugeordnet
- Alle 5 Sekunden leuchtet zufällig eine Lampe auf
- Testperson soll zugeordnete Taste drücken
- Vorfeld: Testpersonen erlernen Zuordnung

Hick-Hyman Gesetz

- Informationsgehalt der Aufgabe („choice“)  
 $H = \log_2(n)$
- Reaktionszeit („reaction time“)  
 $RT = BZ + IV \cdot \log_2(n)$  wobei  $IV$  die Informationsverarbeitungsgeschwindigkeit ist und  $BZ$  die Bewegungszeit
- Reaktionszeit in hierarchischen Menüs folgt dem Hick-Hyman-Gesetz laut Landauer & Nachbar, 1985
- Reaktionszeit bei häufig verwendeten Aktionen folgt dem Hick-Hyman-Gesetz ist aber linear bei selten verwendeten Aktionen laut Sears & Shneiderman, 1994
- Bei Anfängern ist die  $RT$  in einem Menü linear, bei Experten folgt sie dem Hick-Hyman-Gesetz (laut Cockburn & Gutwin, 2008)  
**Beachte:** Bei plötzlichen Menü-Änderungen sind auch Experten wieder Anfänger und müssen das Menü neu erlernen!

4 Post-WIMP

WIMP nimmt eine Anwendungssituation an:

- Benutzer kennt Grundkonzepte von Computern
- Benutzer ist konzentriert auf den Computer
- Bearbeitung alleine und vom Nutzer getrieben
- Hardware = Desktop-PC, Maus, Tastatur, Bildschirm
- Büroumgebung (d.h. gute Beleuchtung, wenig Störeinflüsse)

Diese Voraussetzungen sind in heutigen Anwendungssituationen teilweise nicht mehr gegeben daher wandelte sich das WIMP Konzept zu dem heutigem Post-WIMP Konzept.

4.1 WIMP vs. Post-WIMP

WIMP	Post-WIMP
Metaphors	Interaktion in der realen Welt
Direct Manipulation	Delegierung von Aufgaben an das System
See and Point	Describe and input Command
Consistency	Diversity; angepasst an den Einsatzort (kann nicht immer angewendet werden)
WYSIWYG	Erkennung des Wunsches des Benutzers und der Darstellung einer gleichwertigen Ausgabe
User Control	teilweise gegeben oder Shared Control
Feedback and Dialog	nicht zum Zeitpunkt der Interaktion
Forgiveness	User Model; System weiß, was der Nutzer als Ergebnis haben möchte
Aesthetic Integrity	oft verletzen von See and Point durch Ausblenden von Inhalten
Modelessness	User Model vorhanden, um Verhalten vorherzusagen; mehrere Funktionen für z.B. einen Button

4.2 Touchscreens

Technische Varianten: Kapazitiv, resistiv; Single-touch, Multitouch; Druckstärke

4.2.1 Kapazitive Touchscreens

Funktioniert mit einem elektrischen Feld, in fast allen Smartphones

*Vorteil:* lässt sich mit starkem Glas gut vor äußeren Einwirkungen schützen

4.2.2 Resistive Touchscreens

Funktioniert mit zwei Schichten welche aneinander gedrückt werden

*Vorteil:*

- Bedienung mit jedem Eingabestift möglich
- Mit Handschuhen und Prothesen bedienbar
- Genauer als kapazitive Touchscreens
- Geringe Fertigungskosten

*Nachteile:*

- Nur eingeschränktes Multitouch (Two-touch)
- Ist schlecht lesbar bei Sonneneinstrahlung durch Zusatzschicht
- Die Gestenbedienung, aufgrund des notwendigen Drucks, ist erschwert.
- Verschleiß durch die mechanische Belastung beim

Betätigen

- Unerwünschtes Auslösen beim Transport durch Kontakt mit anderen Gegenständen möglich

5 Entwicklung interaktiver Systeme

5.1 DIN EN ISO 9241-110

DIN EN ISO	WIMP
Aufgabengemessenheit	Aesthetic Integrity, See and Point
Selbstbeschreibungsfähigkeit	Direct Manipulation, Metaphors
Lernförderlichkeit	Metaphors, Feedback & Dialog, Forgiveness
Steuerbarkeit	User Control
Erwartungskonformität	Consistency, WYSIWYG
Individualisierbarkeit	Modelessness
Fehlertoleranz	Forgiveness

5.2 Prinzipien der Entwicklung

- Konzentration auf Benutzer
  - Benutzeranalyse
  - Umfeldanalyse
  - Tätigkeitsanalyse
- Frühes und kontinuierliches Testen
  - Erlebbarkeit der Benutzerschnittstelle in Entwurfsphase fördern
  - Integriertes Design
    - \* Erheben von Feedback zum Zusammenspiel von Interaktionselementen bei der Durchführung typischer Aufgaben der Benutzer
    - \* Solche Test erfordern die volle Verfügbarkeit aller involvierten Interaktionselemente die bei herkömmlicher Implementierung erst spät im Entwicklungsprozess gegeben sind.
    - \* Ganzheitliche Betrachtung der Anwendung
    - \* Jedes Teil der Anwendung sollte entworfen und in seinem Interaktionsverhalten umgesetzt werden
  - Iteratives Design
    - \* Bearbeitung kontinuierlicher Änderungsaufträge als Nebenwirkung von «Frühes und kontinuierliches Testen»
    - \* Streng lineare Entwicklungsmodelle (z.B. Wasserfallmodell) sehen Veränderungen an abgeschlossenen Abschnitten nicht vor
    - \* Prototyping: Erlebbarkeit mit reduzierten Implementierungsaufwand verbinden (Prototyping)
    - \* Dokumentation: Gründe von Änderungen müssen rückverfolgbar sein
  - Empirische Daten zur Bewertung der Qualität einer Benutzerschnittstelle erforderlich

### 5.3 Gebrauchstauglichkeit

"Gebrauchstauglichkeit oder Usability bezeichnet die Eignung eines Produktes bei der Nutzung durch bestimmte Benutzer in einem bestimmten Benutzungskontext die vorgegebenen Ziele (*Wirksamkeit, Qualität der Zielerreichung*), effizient (*Kosten-Nutzen-Relation*) und zufriedenstellend zu erreichen."

- **Usability Testing** Die Messung der Gebrauchstauglichkeit durch Beobachtung bzw. Befragung von Benutzern bei der Durchführung einer Aufgabe im Benutzerkontext.
- **Usability Inspection** Eine Vorhersage der Gebrauchstauglichkeit durch einen Gutachter, der in der Lage ist, Probleme der Benutzer vorherzusagen und die Anwendung analysiert.

## 6 Modelle interaktiver Systeme

### 6.1 Ebenen der Benutzerinteraktion

#### Aktionssprache

- **Konzeptuell:** Objekte, Beziehungen
- **Semantisch:** Abbildung Eingabefolgen auf Funktionen der Anwendung
- **Syntaktisch:** Folgen atomarer Eingaben
- **Lexikalisch:** Atomare Eingaben (Klick, Texteingabe)

#### Präsentationssprache

- **Konzeptuell:** Interaktionsmodell
- **Semantisch:** Form der Ausgabe (Menü)
- **Syntaktisch:** Kombinationen von Interaktionselementen
- **Lexikalisch:** atomare Ausgabelemente (Icons, Text)

*Beispiel:* Kommandozeile:

Eingabe des Benutzers `cd /usr/share` (Aktionssprache) → Ausgabe des Systems `[/usr/share]$` – sowie eine Eingabeausforderung (Präsentationssprache)

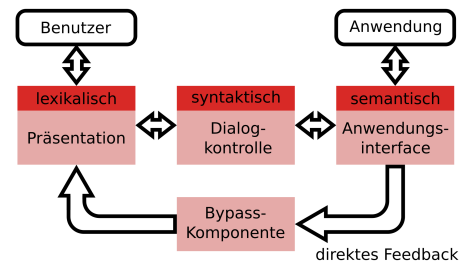
### 6.2 Architekturmodelle

#### 6.2.1 Seeheim-Modell

**Präsentation** Lexikalische Schicht der Benutzerschnittstelle (isolierte I/O-Aktionen)

**Dialogkontrolle** Syntaktische Schicht (bildet Eingaben anhand des Dialogmodells auf Kommandos des Anwendungsinterface ab)

**Anwendungsinterface** Semantische Schicht der Benutzerschnittstelle (Anwendungslogik)

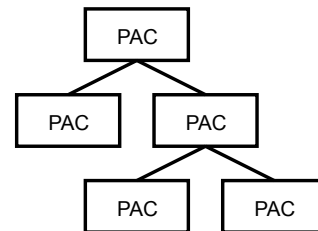


#### 6.2.2 Presentation-Abstraction-Control

**Presentation** Syntax der Interaktion mit der Anwendung (I/O, wie er vom Nutzer wahrgenommen wird)

**Abstraction** Semantik der Interaktion

**Control** Konsistenz von Presentation und Abstraction



*Vorteil:* Wartbarkeit; Austausch von Objekten

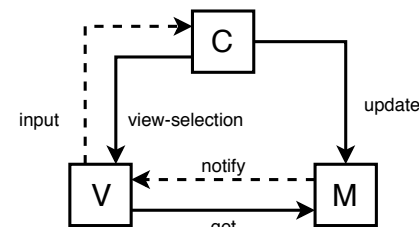
*Nachteil:* Dekomponierung der Interaktion fordernd; Verarbeitungsaufwand

#### 6.2.3 Model-View-Controller (MVC)

**Model** Datenhaltung und Geschäftslogik

**View** Darstellung der Daten

**Controller** Steuerung der Anwendung



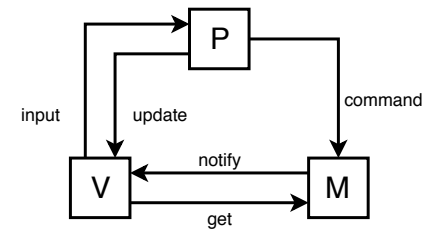
Beobachter-Pattern: View beobachtet Model; Controller kann View austauschen, Kommandos im Modell ausführen, Model ändern

#### Front Controller (Fowler)

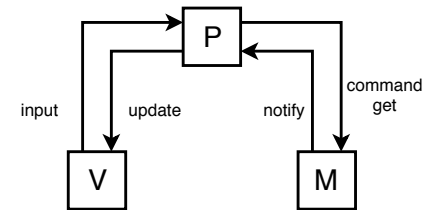
Durch die Steigende Komplexität der Web-Anwendungen wird ein seitenspezifischer Controller eingeführt. D.h. Ein Zentraler Zugangspunkt zuständig für Weiterleitungen, welche mit Hilfe von Tabellenerfolg. Ein Spezifischer Page-Controller generiert im nächsten Schritt die View.

### 6.2.4 Model-View-Presenter

Potol

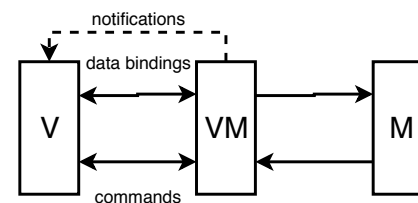


**Fowler** Das Model-View-Presenter Passive View nach Fowler präsentiert die Aufgaben der mit MVC eingeführten Komponenten, dabei beachtet der Presenter das Modell und der Presenter aktualisiert die View.



### 6.2.5 Model-View-Model-Model (MVVM)

Das MVVM Modell ist eine im Wesentlichen aus MVC und MVP abgeleitete Spezialisierung anhand technischer Merkmale von WPF. (Stärkung der Separation of Concerns durch XAML; Beschreibt die Einbindung von Events und Bindings)



### 7.2 Klassifizierung anhand Entwicklungsansatz

#### Black-Box

- **Ansatz**
  - Vordefinierte Komponenten
  - Komponenten werden durch ihre Schnittstelle beschrieben
  - Verständnis der internen Funktionsweise der Komponenten *nicht* erforderlich
- **Anwendung des Frameworks**
  - Objektkomposition
- **Vorteil**
  - Niedrige Einarbeitungszeit

#### White-Box

- **Ansatz**
  - Vordefinierte Komponenten und/oder Codebausteine
  - Verständnis der internen Funktionsweise der Komponenten erforderlich
- **Anwendung des Frameworks**
  - z.B. Erweiterungen des Frameworks durch Ableitungen
  - z.B. Verknüpfen mit Callbacks von Komponenten des Frameworks
- **Vorteil**
  - Hohe Flexibilität

### 7.3 Klassifizierung nach Einsatzbereich

- **Toolkit**
  - Unterstützt die **Erstellung** einer Benutzerschnittstelle durch wiederverwendbare Komponenten (z.B. Menüs, Buttons)
  - Erscheinungsbild und Verhalten der GUI-Komponenten vorgegeben, u.U. anpassbar
- **Interface Builder**
  - Baut auf einem Toolkit auf
  - Unterstützen (i.A. graphischen) **Entwurf** und **Erstellung** der Benutzerschnittstelle ohne Programmieraufwand
- **User Interface Management System (UIMS)**
  - Vermitteln zwischen Benutzer und Anwendungsprogrammen
  - Unterstützen **Entwurf, Erstellung** und **Pflege** von Benutzerschnittstellen
  - Umfassen daher Toolkit und Interface Builder

### 7.4 Prototyping

Prototyping: Realisieren ausgewählte Aspekte eines UI mit dem Ziel Feedback der Benutzer vor der detaillierten Umsetzung eines Interaktionskonzepts einzuholen und so frühes und kontinuierliches Testen zu befördern

### 7.5 Modellbasierte Entwicklung

- **Horizontale Prototypen**
  - Breite Ausdehnung
  - Enthalten vollständige Benutzerschnittstelle aber keine Funktionalität
- **Vertikale Prototypen**

## 7 Frameworks und Tools

### 7.1 Frameworks

Programmiergerüst für die Softwareentwicklung (z.B. .NET Framework)

- Tiefe Ausdehnung
- Enthalten Funktionalität, aber nur einen Teil der Benutzerschnittstelle

7.5.1 Unterscheidung hinsichtlich Übereinstimmung

Low Fidelity	High Fidelity
<ul style="list-style-type: none"><li>• Zweck<ul style="list-style-type: none"><li>– Testen von Ideen, Abläufen</li></ul></li><li>• Vorteile<ul style="list-style-type: none"><li>– Einfaches und schnelles Erstellen</li><li>– Keine Programmierkenntnisse notwendig</li><li>– Änderungen sind schnell herbeizuführen</li></ul></li></ul>	<ul style="list-style-type: none"><li>• Zweck<ul style="list-style-type: none"><li>– Testen von UI-Umsetzung nahe dem finalen Produkt</li></ul></li><li>• Vorteil<ul style="list-style-type: none"><li>– Geringe Abstraktion gegenüber dem finalen Produkt</li><li>– Testen in realer Arbeitsumgebung möglich</li><li>– Zusammenspiel zwischen UI und Anwendung kann getestet werden</li></ul></li></ul>

**Techniken** Storyboards und Interface Flow Diagram (Low Fidelity), Mockups (High Fidelity), Wireframes (Low Fidelity)

7.6 Modellbasierte Entwicklung

Das **Ziel** ist die Flexibilisierung von der Design-Phase (Entwicklungsprozess) und der Runtime-Phase (im Wirkbetrieb)

Ansatz

- Aspekte der Benutzerschnittstelle durch Modelle plattformunabhängig beschreiben
- Modelle als Eingabe für eine automatische Generierung der Benutzerschnittstelle verwenden
- Änderungen der Anforderungen im Modell, nicht im Code
- Vermeiden von Inkonsistenzen zwischen Modell und Code als Folge händischer Überführung

7.6.1 Modelltypen im Cameleon-Reference-Framework

- Domain-Modell
  - Concepts: Konzepte die im Anwendungsgebiet vorkommen
  - Tasks: Aufgaben, die der Benutzer durchzuführen hat
- Context-of-Use-Modell
  - User: Benutzer und Rollen im zu unterstützenden Arbeitsablauf
  - Platform: Technische Möglichkeiten und Grenzen involvierter Geräte; Interaktion von Geräten im Anwendungskontext; Verfügbare Interaktionskomponenten (Presentation Model)

- Environment: Umgebung, in der die Anwendung genutzt wird
- Adaption-Modell
  - Evolution: Zu welcher Benutzerschnittstelle soll gewechselt werden und welches Transition Model soll verwendet werden
  - Transition: Wie soll der Übergang zwischen zwei Benutzerschnittschnellen dem Benutzer kommuniziert werden

## 8 Windows Presentation Foundation

```
1 <Window x:Class = "Zusammenfassung.MainWindow"
2   xmlns = "http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3   xmlns:x = "http://schemas.microsoft.com/winfx/2006/xaml"
4   xmlns:d = "http://schemas.micorsoft.com/expression/blend/2008"
5   xmlns:local = "clr-namespace:Zusammenfassung"
6   Title="MainWindow" Height="350" Width="550"
7   DataContext="{Binding RelativeSource={RelativeSource Self}}"/>
8 <StackPanel>
9   <TextBox Name="Box1"
10     Text="{Binding
11       ElementName=Box2,
12       Path=Text,
13       UpdateSourceTrigger=PropertyChanged}" />
14   <TextBox Name="Box2">
15 </StackPanel>
16 </Window>
```

```
1 public partial class MainWindow : INotifyPropertyChanged
2 {
3   public event PropertyChangedEventHandler PropertyChanged;
4
5   private void NotifyPropertyChanged([CallerMemberName] String name = "")
6   {
7     PropertyChanged?.Invoke(this, new PropertyChangedEventArgs(name));
8   }
9
10  public MainWindow()
11  {
12    InitializeComponent();
13    Binding binding = new Binding("Storage")
14    {
15      Source = this,
16      UpdateSourceTrigger = UpdateSourceTrigger.PropertyChanged,
17      Mode = BindingMode.TwoWay
18    };
19    Box2.SetBinding(TextBox.TextProperty, binding);
20  }
21
22  private string _storage;
23  public string Storage
24  {
25    get
26    {
27      return this._storage;
28    }
29
30    set
31    {
32      if (value != this._storage)
33      {
34        this._storage = value;
35        NotifyPropertyChanged();
36      }
37    }
38  }
39 }
```

### 8.1 Events

```
1 <Button Content="Drueck Mich!" Click="OnButtonClick" />
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

#### 8.1.1 Routed Events

Routed Events pflanzen sich entweder aufwärts oder abwärts in der Hierarchie der *Visual Trees* der GUI fort: Zwischen drin liegende Layout-Manager (**Grid**, **StackPanel**, etc.) bekommen nur dann ein Event, wenn ein enthaltendes Elementen ebenfalls ein Event geschickt bekommt. Bei

≥ 2 sich überlagernden Elementen in der gleichen Ebene wählt das Hit-Testing *immer* das im XAML *zuletzt* deklarierte. Die allgemeinen Event Args heißen **RoutedEventArgs**

- Direkte Events (werden nicht weitergereicht)
  - Ereignisse werden nur von dem Element verarbeitet, bei dem sie auch aufgetreten sind (konform mit .NET)
- Bubbling-Events
  - Event wird zuerst an den Handler des Quellelements weitergereicht. Dann wird sich dieses Weiterreichen elementweise bis zur Wurzel fortgesetzt (meist Window)
- Tunneling Events
  - Hier ist es umgekehrt wie beim Bubbling. Die Ereigniskette beginnt beim Wurzelement, d.h. Window oder Page. Nachfolgend der Handler des untergeordneten Elements usw.
- Bubbling/Tunneling stoppen: **args.Handled = true;**

### 8.2 Attribut-Element-Syntax

```
1 <Button Name="example" Content="Drueck Mich!" Height="20">
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

### 8.3 Bindings

Bindungen binden immer nur auf **Properties**, nicht auf Member-Variablen, und der angegebene *Path* muss public sein.

#### 8.3.1 Richtungsbestimmung

- *TwoWay*: Zweibegebindung
- *OneWay*: Von der Quelle zum Ziel
- *OneWayToSource*: Von dem Ziel zur Quelle
- *OneTime*: Eine einmalige *OneWay* Bindung

#### 8.3.2 Data-Context

```
1 public MainWindow()
2 {
3   InitializeComponent();
4   DataContext = this;
5 }
6
7 public string StudentName { get;set; }
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
```