

# **Interaktive Systeme**

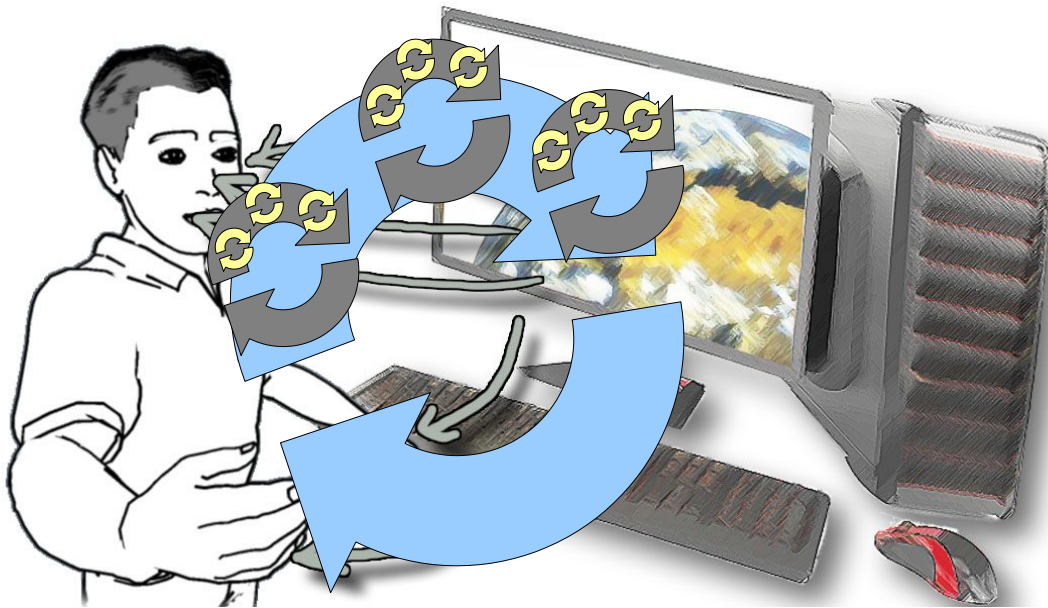
## **6. Interaktionsschleife**

Prof. Dr. Eckhard Kruse

DHBW Mannheim

# Interaktion - Regelkreise

**Interaktion = Aktion (Mensch) + Reaktion (Computer)**



Regelkreise der Interaktion erfolgen auf verschiedenen, hierarchisch aufeinander aufbauenden Ebenen mit unterschiedlichen Zykluszeiten.



### Übung

#### 6.1 Interaktionsregelkreise

Betrachten Sie Regelkreise in interaktiven Anwendungen:

- a) Listen Sie verschiedene Arten der Interaktionen vom Millisekunden- bis Minutenbereich
  - Was ist das Ziel der Interaktion?
  - Wie erfolgt die Rückmeldung / Rückkopplung?
  - Wie lange dauert die gesamte Interaktion?
  - Wie schnell sollte die Rückmeldung idealerweise erfolgen? Wie langsam dürfte sie gerade noch für ein erträgliches Arbeiten sein?
  - Wo gibt es Regelkreishierarchien?
- b) Wie schnell agiert/reagiert der Mensch in diesen Regelkreisen?
  - Nennen Sie Einflussfaktoren
- c) Betrachten Sie Interaktionsregelkreise in technischen Systemen außerhalb der Computerwelt
  - Nennen Sie Beispiele. Wie beurteilen Sie das Zeitverhalten?

**Responsiveness\*** ("Antwortfähigkeit") bezeichnet bei interaktiven Systemen die Eigenschaft, auf Eingaben des Anwenders so zeitnah zu reagieren, dass ein effizientes / angenehmes Arbeiten möglichst ohne Wartezeiten möglich ist.

*\* Achtung: „Responsive Design“ → andere Bedeutung*

- Sinnvolle Mindestantwortzeiten hängen von vielen Faktoren ab:
  - Art der Interaktion, Anwendung, Stand der Technik: Erwartungshaltung
- Typische Forderung: Einhalten von Obergrenzen für Antwortzeiten
  - wichtig z.B. in Anforderungsspezifikation
  - Worst Case / Average Case?

Beispiele?

Beispiele?

Inwiefern unterscheiden sich Maus und Tastatur bei der Handhabung von Eingabeverzögerungen? Welche Antwortzeiten halten Sie für noch tolerierbar?

- Jacob Nielsen: 0.1 Sekunde, 1 Sekunde, 10 Sekunden...

# Rückkopplung Grundsätze

## Grundsätze für Rückmeldungen

- schnell / rechtzeitig
- angemessene Sichtbarkeit
  - erkennbar, aber nicht störend oder ablenkend
  - Fehler und Probleme besonders hervorheben
- Deutliche Anzeige des Modus/Betriebszustands
- Lenken der Aufmerksamkeit
  - z.B. Hervorheben von selektierten Objekten
  - Visualisierung des Abschlusses von Aufträgen
- Geeigneter Sinneskanal: Akustisch oder optisch (fast immer)
- Anwender berücksichtigen
  - Vorerfahrungen: Anfänger benötigt mehr Rückkopplung als Experte
  - Sonstiges: z.B. Seh- / Hörvermögen / Motorik → Accessibility (s. z.B. Mauseinstellungen → schnell / langsam)

*System sollte stets „Lebenszeichen“ geben.*

*UI Design Pattern: „**Make it direct**“: Input, wo der Output ist.  
Beispiele? Gegenbeispiele?*

### Übung

#### 6.2 Schnell – und was noch?

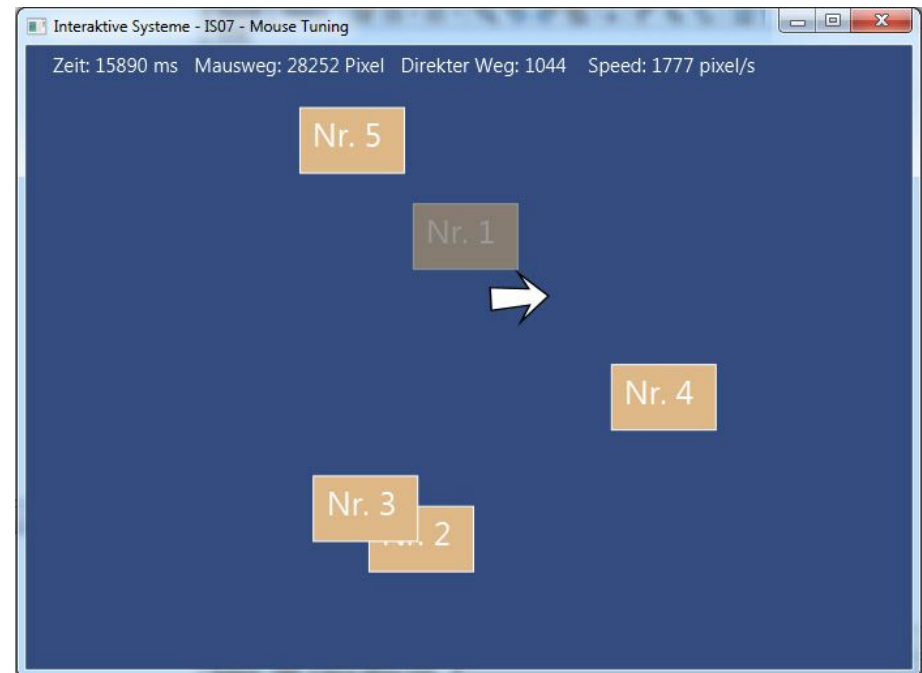
Schnelle Antwortzeiten alleine reichen nicht:

- a) Drehen Sie Ihre Maus um 180 Grad (so dass die Tasten vorne liegen).  
→ Führen Sie typische Aktionen am Computer durch, d.h. bewegen Sie den Mauszeiger gezielt zu verschiedenen Menüs, Buttons usw.
- b) Schalten Sie Ihre Tastatur von deutsch auf englisch um und tippen Sie einen längeren Text.
- c) Drehen Sie Ihre Tastatur um 180 Grad und tippen Sie einen Text.
- d) Interpretieren Sie Ihre Erfahrungen. Was ist passiert? Was für Folgerungen ergeben sich für die Interaktion?
- e) Welche Erfahrungen würde jemand, der nie zuvor mit einem Computer gearbeitet hat, bei dieser Übung machen?

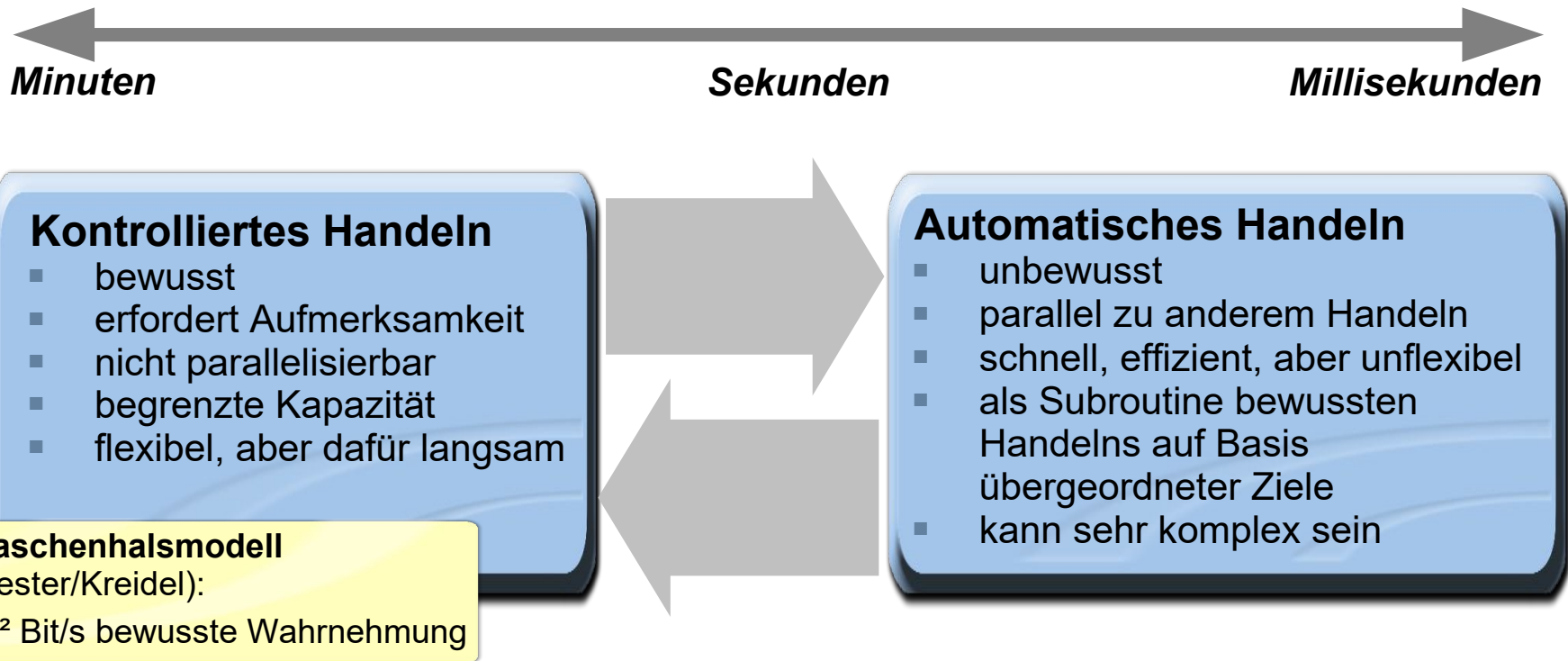


### Maus-Tuning

- Experimentieren Sie mit den Parametern zur Mausbewegung und Darstellung.
- Wie lässt sich maximale Performance erreichen?
  - Darstellung des Mauszeigers?
  - Geschwindigkeit, Beschleunigung?
  - Größe der GUI-Elemente? (Abhängigkeit Größe ↔ Geschwindigkeit)
- Messen Sie Zeit und Treffgenauigkeit beim Klicken von GUI-Objekten.
- Wie wirkt sich ungewohnte Mausbedienung aus, z.B. Spiegelung der x-/y-Koordinaten? Wie sieht dann der Lernfortschritt bei der Bedienung aus?
- Recherchieren Sie nach **Fitts' Law** - Wie ließe sich dieses mit dem Beispielprogramm überprüfen?
- Wie ließe sich mit dem Programm eine Button-Anordnung testen/optimieren?
- Wie könnte mit den Nielsen-Zeitangaben (s. Folie 4) experimentiert werden?



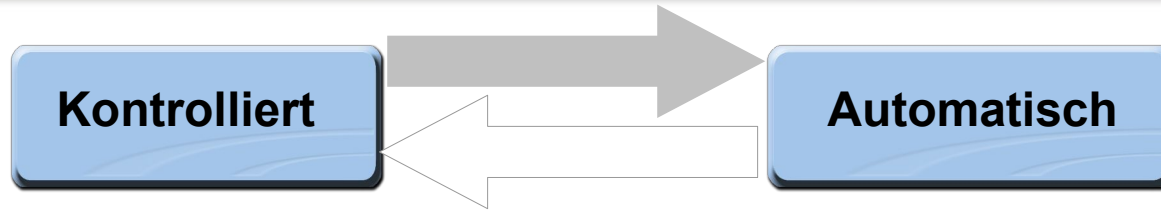
# Kontrolliert vs. automatisch



- a) Beispiele für beide Handlungsweisen?
- b) Wann/wie kommt es zu Übergängen zwischen den Handlungsweisen?
- c) Folgerungen für die Gestaltung interaktiver Systeme?



# Automatisches Handeln



## Automatisches Handeln soll leicht erlernbar sein

- Gleiche/ähnliche Abläufe bei ähnlichen Vorgängen
  - Kontextabhängigkeit gefährdet Konsistenz
- Wiederverwendung bekannter Paradigmen
  - z.B. rechte Maustaste für Kontextmenü
  - Applikations-/System-übergreifend?
- Bekanntes aus der Technik-/Alltagswelt nachahmen
  - Buttons, Radio buttons, Fenster, drag&drop ...
  - aber: Einfachheit geht vor Sinn
- Hinweise zu Vereinfachungen/Shortcuts
  - z.B. Tastaturkürzel in Menüs

Beispiele aus  
interaktiven Systemen:  
Was funktioniert gut /  
schlecht?

## Automatisches Handeln soll effizient sein

- Handlungsvorgänge auf Geschwindigkeit, Effizienz hin optimieren (z.B. Keyboard shortcuts)
- Parallelisierung ermöglichen (z.B. Tastenkombinationen mit nur einer Hand, verschiedene Eingabekanäle nutzen, ...)

# Die richtige Mischung

Der Entwurf von Interaktionsmechanismen sollten die verschiedenen Handlungsweisen gezielt berücksichtigen und die Übergänge sinnvoll gestalten.

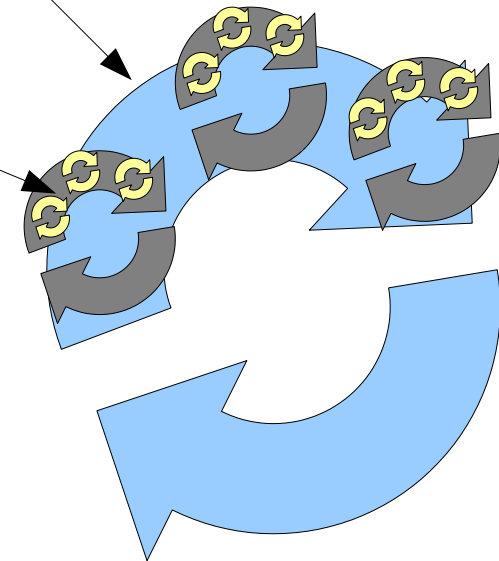
Wie? Beispiele?

## Typisch: Mischung aus automatischem und bewussten Handeln

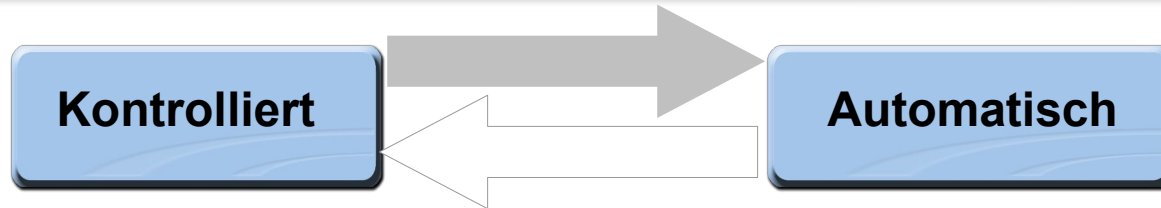
- Bewusstes Handeln für übergeordnete Ziele
- Automatisches Handeln ist nebenläufig möglich:
  - für untergeordnete Ziele
  - parallele Aktionen

## Sensomotorik:

- Direktes Zusammenspiel von Sinneswahrnehmung und Motorik
- vollständig automatisiert + unbewusst
- Grundlage für übergeordnetes automatisches und bewusstes Handeln
- Beispiele:
  - Hand-Auge-Koordination (z.B. beim Greifen)
  - Bewegungsabläufe des Körpers (z.B. beim Sport)



Interaktionsschleife



## Lernvorgang: Aus kontrolliertem wird automatisches Verhalten.

- Wiederholung → Abläufe werden schneller und automatisch
  - auch sehr komplexes Verhalten
- Problem: Nicht immer führt ein Lernprozess kontinuierlich zum Optimum:
  - Um schneller/effizienter zu werden muss man u.U. vorübergehend langsamer werden
- Ideal: Anwender lernt das Automatisieren in interaktiven Systemen mühelos nebenbei.

Beispiele?



## Lernen-Exkurs...

- Psychologie: Belohnungsaufschub + Impulskontrolle
- ist verlässlicher Indikator für späteren akademischen Erfolg!

# Einfach Lernen?

## Lernansätze:

### ■ Häufiges Wiederholen

- ok, wenn es sowieso, mit sinnvoller Zielsetzung getan wird (z.B. Fremdsprache im Ausland sprechen)
- schlecht/mühsam, wenn es Selbstzweck ist (z.B. Vokabeln zu Hause auswändig lernen)

### ■ Beziehungen herstellen

- Assoziationspaare, z.B. Absicht → Handlung:
  - Befehl (Menü, Tastatur) → Programmfunktion
- Einheitlichkeit?! find ↔ search, exit ↔ quit, UI deutsch ↔ englisch ...
- → **Konsistenz anstreben!**

### ■ Wiederkennen

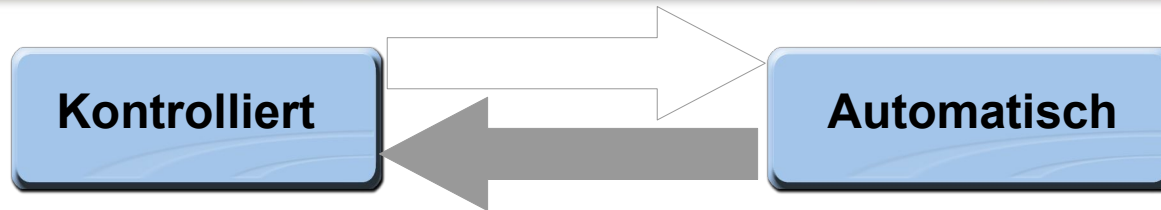
- *"Sieht aus wie ein Knopf/Button → da kann ich wohl draufdrücken"*
- Computer ahmt Wirklichkeit nach (GUI Metaphern, Wii als Eingabegerät...)
- Außerdem: Erweiterung der Realität (Fenster mit Scrollbalken...)

## Aber: Gewohnheit ist oft entscheidender als Sinn:

- Sinn spielt nur beim Lernen / bei neuen Dingen eine Rolle
- *"Das haben wir schon immer so gemacht..."*

Beispiele?  
Vorteile/Nachteile?

# Kontrolliertes Handeln



## Bewusster Einsatz von kontrolliertem Handeln

- Durchbrechen von automatischem Handeln, wenn bewusstes Handeln wichtig ist

Beispiele?

## Kontrolliertes Handeln soll effizient sein

- Aufmerksamkeit auf die notwendigen Handlungen richten:
  - optisches Hervorheben, akustische Signale
  - Modale Dialogfelder (z.B. Fehlermeldungen) → keine anderen Aktionen möglich
  - ...
- Zu viele Alternativen, Optionen verwirren
  - z.B. Bestellvorgang: Sequenzielles Abarbeiten

# Modus

Betriebsmodus/Interaktionsmodus: Je nach Modus führen gleiche Eingaben zu unterschiedlichem Systemverhalten.

**Beispiele? Warum sollte man so etwas wollen?**

Zu viele  
Eingabeelemente?





# Modus

Betriebsmodus/Interaktionsmodus: Je nach Modus führen gleiche Eingaben zu unterschiedlichem Systemverhalten.

Bitte wählen Sie den Betriebsmodus des Multifunktions-Pedals:

- Gas geben
- Kuppeln
- Bremsen

Gute Idee?

Und doch ist es in technischen Systemen weit verbreitet...

Gute/schlechte Beispiele?

Warum sollte man in Software / GUIs so etwas wollen?



### Übung

#### 6.3 Modus

Interaktions-/Betriebsmodi spielen in interaktiven Anwendungen eine wichtige Rolle.

- a) Nennen Sie Beispiele von verschiedenen Betriebsmodi in Betriebssystemen und Standardanwendungen.
  - Wie viele alternativ einsetzbare Modi werden jeweils angeboten?
- b) Nennen Sie Vor- und Nachteile von Modi.
- c) Wie werden Betriebsmodi dargestellt und umgeschaltet ?
- d) Welche Aktionen sollten immer einheitlich/modusunabhängig ausgeführt werden?
- e) Erstellen Sie eine Empfehlungsliste für die Entwicklung/Bereitstellung von Interaktionsmodi.

## Modus-Verwirrung...?

Some accidents and incidents in highly automated aircraft have involved pilots not being able to overcome the protection limits or the pilots not being aware that the protection functions were in force. For example, the pilots during one A-320 approach disconnected the autopilot while leaving the flight directors and the autothrust system engaged. Under these conditions, the automation provides automatic speed protection by preventing the aircraft from exceeding upper and lower airspeed limits:

At some point during the approach, after flaps 20 had been selected, the aircraft exceeded the upper airspeed limit for that configuration by 2 kts. As a consequence, the automation intervened by pitching the airplane up to reduce airspeed back to 195 kts. The pilots, who were not aware that the automatic speed protection was active, observed the uncommanded automation behavior. Concerned about the unexpected reduction in airspeed at this critical phase of flight, they rapidly increased thrust to counterbalance the automation. As a consequence of this sudden burst of power, the airplane pitched up to about 50 degrees, entered a sharp left bank, and went into a dive. The pilots eventually disengaged the autothrust system and its associated protection function and regained control of the aircraft [SW95].

aus: Nancy G. Leveson et. al., *Analyzing Software Specifications for Mode Confusion Potential*

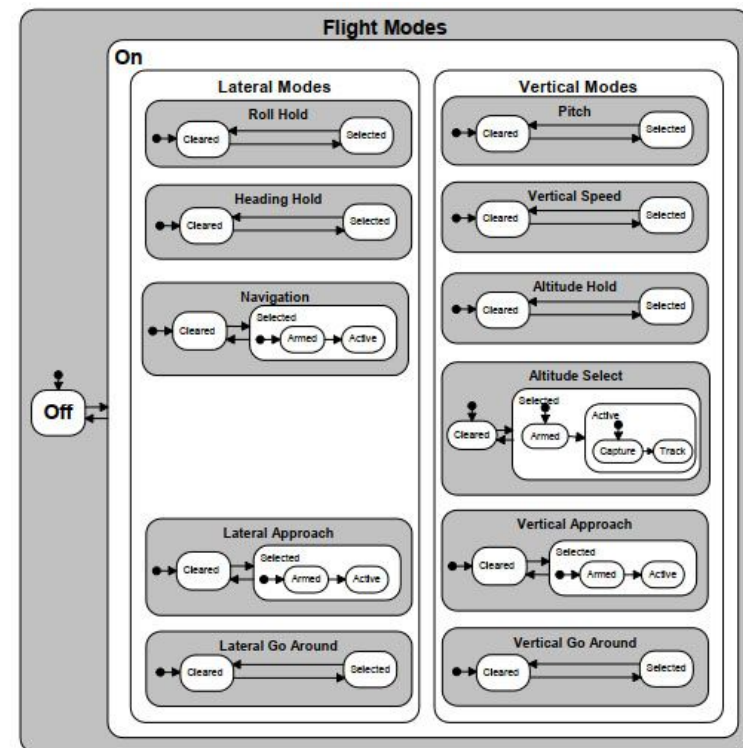
und: Sarter, N. D. and D. Woods, „How in the world did I ever get into that mode? Mode error and awareness in supervisory control. Human factors 37, 5-19.

# Mode confusion

On September 14, 1993, a Lufthansa Airbus A320 overran the runway after landing at Warsaw Airport, killing a crew member and a passenger, and injuring 54. The A320 control logic required the airplane to be settled on both main landing gears before the brakes, ground spoilers and thrust reversers can be activated. The airplane did not settle onto its second main landing gear for nine seconds, at which point it was still traveling at 154 knots -- 20 knots above normal landing speed -- with only 1000 m of runway left.

Quelle: H. Kopetz: Real-Time Systems

Modus-Thema?  
Vergleich zu SW-  
Anwendungen?



Mode confusion analysis of a flight guidance system using formal method, Joshi, Miller, Heimdahl, 2003

### Übung

#### 6.4 Modus - Entwurf

Ein Zeichenprogramm soll eine Palette von Werkzeugen mit unterschiedlicher Funktionalität anbieten. Für die Werkzeuge gibt es verschiedene einstellbare Parameter und Darstellungsweisen. Werkzeuge sind in Werkzeuggruppen eingeteilt, die gemeinsame Parameter (z.B. Stiftbreite) haben.

- a) Erstellen Sie einen objektorientierten Entwurf für die Auswahl und Bedienung der verschiedenen Werkzeuge:
  - Welche Klassen, Abhängigkeiten und Attribute gibt es?
  - Wie und wo realisieren Methoden die erforderliche Funktionalität?
- b) Was passiert beim Umschalten des Modus? Inwieweit bleiben die eingestellten Parameter erhalten?



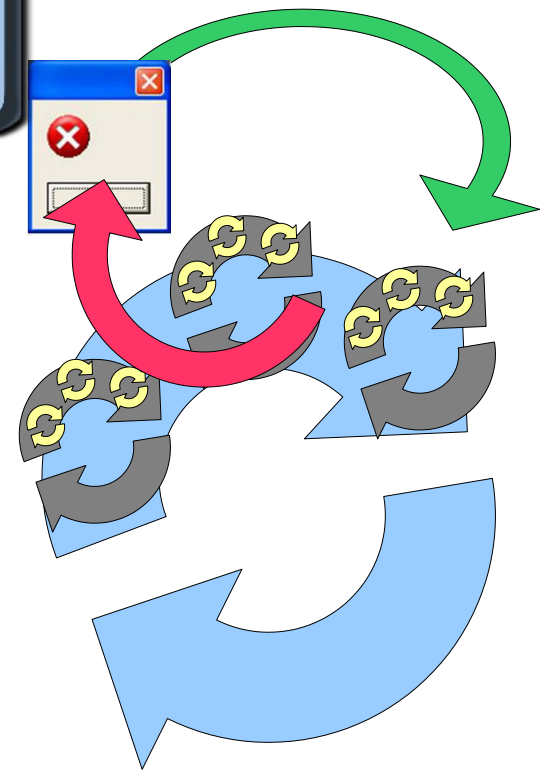
# Fehler

Fehler in interaktiven Systemen:  
Das (Teil-)Ziel einer Handlung wurde nicht erreicht.  
(→ Ohne Ziel kein Fehler)

Unerwartete Zustände bzw. Fehler unterbrechen automatisierte Interaktionsabläufe.

Wichtig:

- möglichst schnell von der automatisierten auf die kontrollierte Handlungsebene
- Möglichkeit, Fehler schnell und einfach „ungeschehen“ zu machen oder die Auswirkungen zu begrenzen (→ Undo)
- Schnelle Rückkehr in effiziente automatisierte Arbeitsweise



Interaktionsschleife

Entdeckung → Diagnose → Korrektur



### Übung

#### 6.5 Fehler in der Interaktion

- a) Warum ist es wichtig, die Interaktion möglichst schnell von der automatisierten auf die bewusste Handlungsebene zu heben? Was kann andernfalls passieren? Nennen Sie Beispiele.
- b) Nennen Sie Beispiele für gute und schlechte Mechanismen zur Fehlerbehandlung. Beziehen Sie sich auf die drei Schritte Entdeckung, Diagnose, Korrektur.
- c) Erstellen Sie eine Liste von Empfehlungen zur Fehlerbehandlung, benennen Sie Bearbeitungsschritte und Ausgaben im Rahmen der drei Schritte.
- d) Vergleichen Sie die bisherigen Überlegungen mit den Konzepten der Fehlerbehandlung im Material Design.
- e) Wie hoch schätzen Sie den ungefähren Zeitanteil für die Fehlerbehandlung in interaktiven Anwendungen?

- **Angaben in Fehlermeldungen**
  - Was ist falsch? Von welcher Art ist der Fehler?
  - Welche Ursache hat der Fehler? Bis wo ist die Bearbeitung fehlerfrei?
  - Welche Korrekturmaßnahmen können getroffen werden?
- **Wann und wie Fehlermeldungen zeigen**
  - Eingabealternativen bei kleiner Anzahl, Aufrufmöglichkeit weiterer Hilfe
  - so schnell wie möglich, höchstens bis zur Behebung des Fehlers
  - konsistent an bestimmtem Platz, vorne, ggf. nicht verdeckbar, andere Funktionen sind gesperrt (modaler Dialog)
- **Fehlerkorrektur durch das System**
  - System verfügt über Lösungsmöglichkeiten
  - Anwender wählt ggf. zwischen (wenigen) Alternativen („Retry“ ...)
  - Autokorrektur (was ist wirklich richtig...?)
  - Hoffentlich auch abschaltbar (s. z.B. Textverarbeitungsprogramme)
- **Fehlerkorrektur durch den Anwender**
  - Mittel zur Fortsetzung des Dialogs erforderlich
  - Werkzeuge zur Fehleridentifizierung und Korrektur
  - Abbruchmöglichkeit: Cancel-Button, Undo-Funktion (am besten multi-Level)
  - Bearbeitung der fehlerhaften Daten statt erneuter Eingabe

# Fehlerarten

## Sensomotorische Fehler

- Vertippen, falsche Mausclicks, ...

## Gewohnheitsfehler

- Scheinbar ähnliche Situationen erfordern verschiedene Handlungen (z.B. Programme mit verschiedenen Tastatur-Shortcuts)
- Neue Situation erfordert Umlernen der Gewohnheiten

## Unterlassensfehler

- Überspringen von Teilen durch gedankliches Vorseilen
- Weglassen am Ende des Handlungsmusters
- besonders bei Unterbrechungen oder bei Wartezeiten

## Erkennensfehler

- Übersehen, Überhören
- Ablenkung durch andere Prozesse / andere Wahrnehmungen

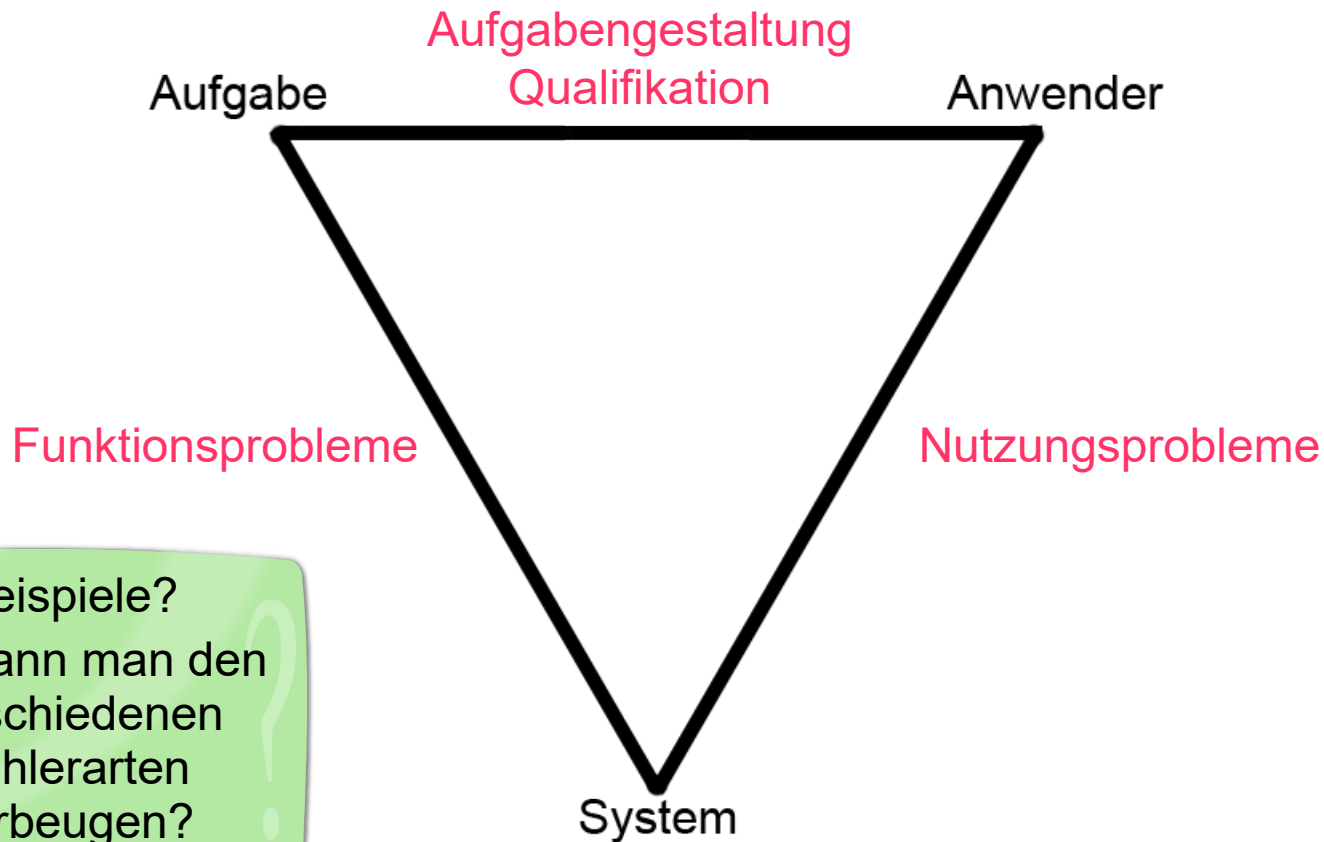
## Fehler bei bewussten Denkprozessen

- Denkfehler: z.B. Außerachtlassen von Nebeneffekten, Wahl ungeeigneter Werkzeuge
- Wissens-/Merkfehler: z.B. Fehlende Kenntnis / Vergessen von erforderlichen Handlungsschritten, falsche mentale Modelle
- Urteilsfehler: z.B. Falsche Einschätzung von Ergebnissen, erforderlichen Handlungen

Beispiele?  
Wie kann man  
diesen Fehlern  
vorbeugen?

# Mismatch

Mismatch-Konzept (Frese und Zapf, 1991):  
Einem Fehler liegt immer ein "Nicht-Passen" zugrunde.

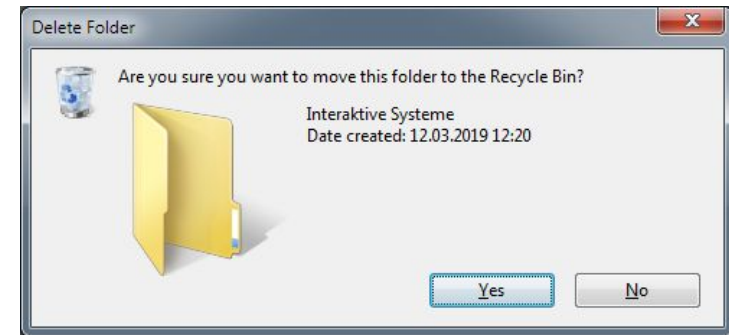


Beispiele?  
Wie kann man den  
verschiedenen  
Fehlerarten  
vorbeugen?

# Fehlervermeidung

## Hauptziel: Minimierung von Gewohnheitsfehlern, Unterlassensfehlern, Erkennensfehlern, sensomotorischen Fehlern

- Schlüssige Gestaltung der Benutzungsoberfläche
  - Konsistenz + Erwartungskonformität  
(→ was erwartet der Anwender)
- Ergonomische Gestaltung  
(z.B. Größe und Lage der Bedienelemente)
- Einschränkung der Freiheitsgrade  
(erlaubte Eingaben)
- Sicherheitsabfragen und Warnungen



Web

Meinten Sie: [interaktive systeme](#) Die ersten 2 angezeigten Ergebnisse

## Minimierung von Fehlern bei bewussten Denkprozessen:

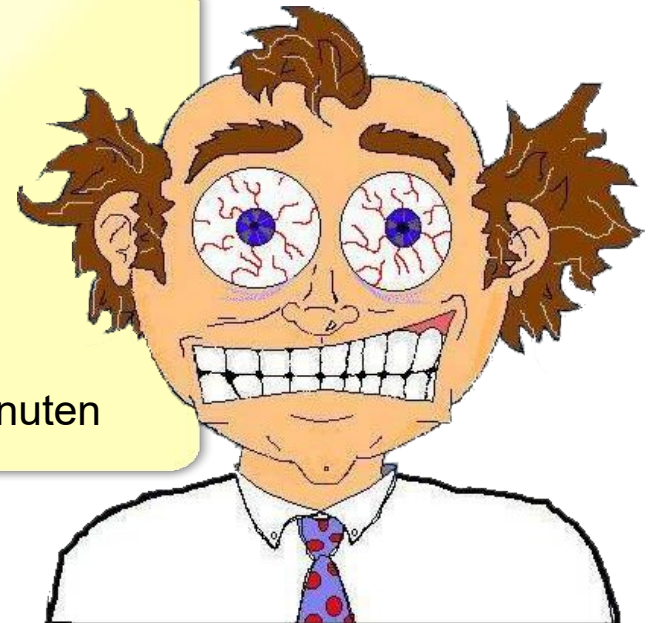
- Denk-/Wissensfehler: Indirekt durch Hilfen, Tutorials, Wizards ...
- Merk-und Vergessensfehler: Reduzierung der erforderlichen Gedächtnisleistung (z.B. Menüs statt Kommandos, Tooltips)
- Urteilsfehler: Aussagekräftige Eingabedialoge, GUI-Gestaltung, Online-Hilfen

# Fehler und Emotionen

Fehler kosten nicht nur Zeit, sondern können auch zu Motivationsverlust und psychischen Belastungen (Ärger, Frustration, Stress) führen.

Anwender zeigten bei der Fehlerbewältigung überdurchschnittlich starke Emotionen (Projekt FAUST):

- 8% bei sofortiger Korrektur des Fehlers
- 15% bei einer Fehlerkorrekturzeit bis 2 Minuten
- 34% bei einer Fehlerkorrekturzeit bis 5 Minuten
- 36% bei einer Fehlerkorrekturzeit bis 10 Minuten
- 57% bei einer Fehlerkorrekturzeit von mehr als 10 Minuten



Bildquelle: [http://auerhouse.net/images/Stress\\_Stinks.jpg](http://auerhouse.net/images/Stress_Stinks.jpg)