

Gesetze, Normen, Designprinzipien und Richtlinien

Ilhan Aslan, Chi Tai Dang, Björn Bittner, Katrin Janowski,
Elisabeth André



Human Centered Multimedia

Institute of Computer Science

Augsburg University

Universitätsstr. 6a

86159 Augsburg, Germany

- Grundlagen:
 - Designprinzipien
 - Normen und Gesetze
 - Richtlinien und Empfehlungen
- Grundlegende Designprinzipien mit Anwendungsbeispielen
- „Pakete“ von Designregeln

Gesetze, Normen, Designprinzipien und Richtlinien - **Grundlagen**

Ziel:

Beschreibung und Dokumentation von **bewährtem Wissen** über die **Gestaltung** von z.B. Systemen, Benutzeroberflächen, Dialogverläufen und Webseiten, die **Usability-Ziele** erfüllen

Forschungsfragen:

- Welche Regeln sind grundsätzlicher Natur, welche eher speziell?
 - Wie ist das Wissen zu beschreiben und zu dokumentieren, damit es von Softwareentwicklern aufgegriffen und umgesetzt werden kann?
- **Designprinzipien, Normen, Gesetze und Richtlinien für das Design und die Usability von Systemen**

Beispiele:

- DIN EN ISO 9241 (seit 1998)
 - Teil 11, Richtlinien zur Gebrauchstauglichkeit
- Jakob Nielsen's Top Ten Mistakes in Web Design (www.useit.com)
- Macintosh Human Interface Guidelines, Apple Computer
- Microsoft Windows User Experience, Microsoft Press
- Java Look and Feel Design Guidelines, Sun Microsystems
- Google: „Design Guidelines“
- u.v.m.

Verwendung des Wissens:

- **Konstruktiv:**
 - Entscheidungshilfe:
 - beim Entwurf/Entwicklung eines Systems
 - bei der Produktion von Designlösungen
- **Analytisch:**
 - Überprüfung und Bewertung von Designlösungen bzgl. der Einhaltung von:
 - Zielen und Anforderungen
 - vorgegebenen Rahmenbedingungen
 - Corporate Identity
 - Common Look&Feel
 - Checklisten zur Systemevaluation
(vgl. Foliensatz - Analytische Evaluation)

- **Prinzipien / Normen / Gesetze:**

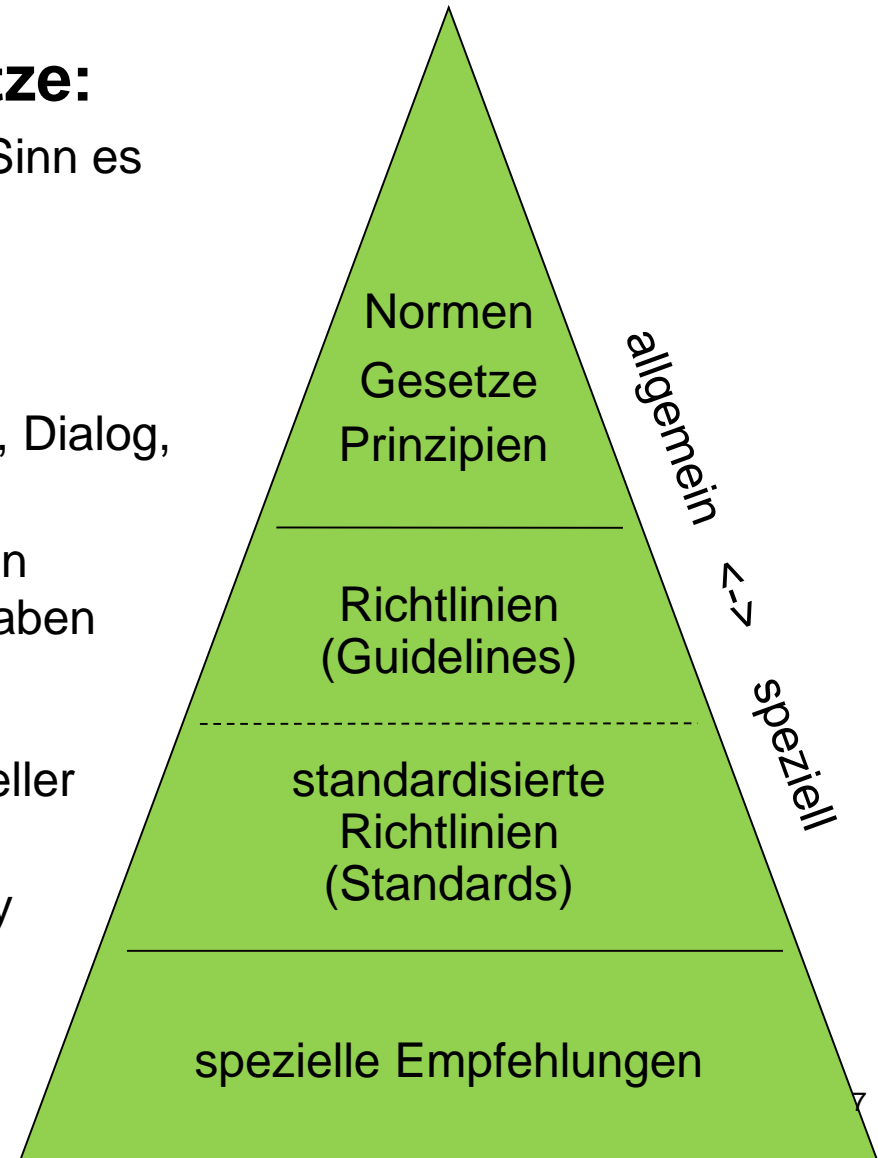
- allgemein formulierte Ziele, deren Sinn es ist, die Entscheidungsfindung im Designprozess zu unterstützen

- **Richtlinien**

- für ein bestimmtes Feld (GUI, Web, Dialog, Mobile usw.) formulierte Regeln
- u.a. für die konkrete Umsetzung von Prinzipien für konkrete Designaufgaben

- **Spezielle Empfehlungen**

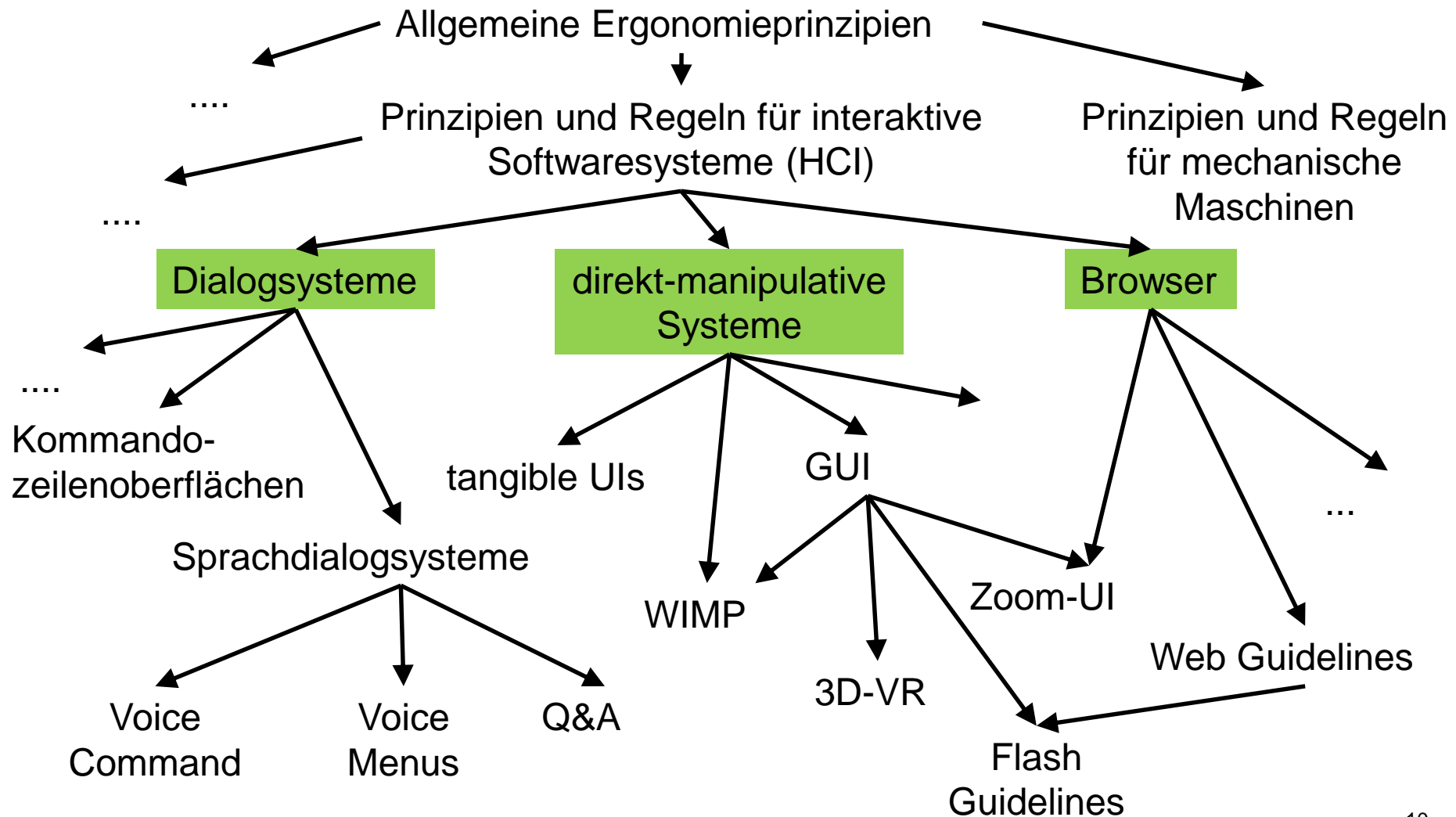
- weitere Regeln zur Erfüllung spezieller Anforderungen, z.B.
 - Wahrung der Corporate Identity
 - kultur-spezifische Präferenzen
 - Style Guides



- auf Erfahrung:
 - „gelernte Lektionen“ aus Problemen mit Interfaces
- auf empirischen Untersuchungen:
 - Experimente zur Erschließung des Wesens und der Leistungsfähigkeit menschlicher Wahrnehmung und Informationsverarbeitung
 - Beispiele:
 - Auflösung (z.B. Helligkeits- / Farbkontraste)
 - Verarbeitung von Bildern vs. Sprache
 - Gedächtnisleistung



- auf Modellen und den daraus abgeleiteten Vorhersagen:
 - Vorhersagen über die Bewältigung spezifischer Aufgaben basierend auf Annahmen, wie Menschen allgemein Information verarbeiten
- Andere Quellen:
 - Analogieschlüsse: Übertragen von Regeln auf neue Anwendungsgebiete
 - Intuition/Introspektion: „Was mir gefällt ist auch für andere gut“
 - ...



Warum Normen und Gesetze?

- Unterstützen Projektverantwortliche bei der Gestaltung von Schnittstellen
 - Kann bei der Einhaltung zu einer grundlegenden Usability der Nutzerschnittstelle führen
 - Führt durch die Übermittlung von Hinweisen, Richtlinien und Erfahrungen zu einer gleichen oder ähnlichen Gestaltung von Nutzerschnittstellen
- Weniger Aufwand bei der Einarbeitung in neue Systeme

Eigenschaften von Normen und Gesetzen

- Beschreibungen sind oft sehr allgemein.
- Konkrete Bestimmungen gibt es oft nicht.
- Aber oft werden Empfehlungen abgegeben.

Wer normt?

- DIN – Deutsches Institut für Normung
- CEN – European Committee for Standardization
- CENELEC – European Committee for electrotechnical Standardization
- ETSI – European Telecommunications Standards Institute
- ...

- **DIN EN ISO 9241-11** (Anforderungen an die Gebrauchstauglichkeit)
 1. Effektivität zur Lösung einer Aufgabe
 2. Effizienz der Handhabung des Systems
 3. Zufriedenheit der Nutzer einer Software
- **DIN EN ISO 9241-110** (Grundsätze der Dialoggestaltung)
 1. Aufgabenangemessenheit (Taskunterstützung)
 2. Selbstbeschreibungsfähigkeit (Transparenz)
 3. Steuerbarkeit (Nutzerkontrolle)
 4. Erwartungskonformität (Konsistenz)
 5. Fehlertoleranz (Fehlerkorrektur)
 6. Individualisierbarkeit (Anpassbarkeit)
 7. Lernförderlichkeit (Unterstützung)

DIN EN ISO 9241-110: Grundsätze der Dialoggestaltung

Ziel 1: Aufgabenangemessenheit

- In welchem Maß wird der Nutzer unterstützt, um seine Aufgabe effizient und effektiv umzusetzen. (geeignete Funktionalität, Minimierung unnötiger Interaktionen)

Ziel 2: Selbstbeschreibungsfähigkeit (Transparenz)

- Das Dialogsystem soll jeden Dialogschritt durch Hilfen bzw. Rückmeldungen unmittelbar verständlich machen, wenn Nutzer entsprechende Informationen verlangen.

Ziel 3: Steuerbarkeit

- Nutzer soll in der Lage sein, den Dialogablauf bis zum Ziel der Interaktion selbst zu beeinflussen.

Ziel 4: Erwartungskonformität (Konsistenz)

- Bisherige Kenntnisse aus Arbeitsabläufen, Ausbildung und Erfahrung der Nutzer sowie allgemein anerkannte Übereinkünfte lassen auf Systemverhalten schließen

Ziel 5: Fehlertoleranz

- Sicherstellen, dass erzielt Zwischenresultat trotz fehlerhafter Eingaben mit minimalen oder ohne Korrekturen verwertbar und weiter bearbeitbar ist.
- z.B. vermeide Notwendigkeit des Neubeginns nach Systemabsturz

Ziel 6: Individualisierbarkeit

- Möglichkeit, das System und seine Bedienung an individuelle Belange und Fähigkeiten des Nutzers anpassen zu können (Stichwort: Anfänger/Experte, Adaptierbarkeit, Kontext)

Ziel 7: Lernförderlichkeit

- Unterstützung des Nutzers beim Erlernen der Systembedienung, z.B. durch sinnvolle Online-Hilfe und gute Bedienungsanleitungen bzw. Tutorials
- Verwendung von geeigneten Metaphern
- Ziel: Minimale Einlernzeit

- **DIN EN ISO 14915:** Software-Ergonomie für Multimedia-Benutzerschnittstellen
 - Berücksichtigt Aspekte der menschlichen Wahrnehmung, Kognition, Aufmerksamkeit und Kommunikation
 - Drei Teile:
 1. Gestaltungsgrundsätze und Rahmenbedingungen
 2. Multimedia-Navigation und Steuerung
 3. Auswahl und Kombination von Medien
 - Teil 1 ergänzt die 7 Grundsätze der DIN EN ISO 9241:
 1. Eignung für das Kommunikationsziel
 2. Eignung für Wahrnehmung und Verständnis
 3. Eignung für die Exploration
 4. Eignung für die Benutzungsmotivation

- **DIN EN ISO 13407:** Benutzerorientierte Gestaltung von interaktiven Systemen
 - Beschreibt Vorgehensweisen zur Einhaltung von ISO 9241
 - Entwicklungsprozess wird in 4 Teilaktivitäten unterteilt:
 1. Nutzerkontext verstehen
 2. (Benutzer-)Anforderungen spezifizieren
 3. Entwerfen von Lösungen
 4. Beurteilen von Lösungen

- **VDI-Richtlinie 5005:** Software-Ergonomie in der Bürokommunikation
 1. Kompetenzförderlichkeit:
 - Ermöglichte kompetenten Umgang mit dem System
 - Fördere die Handlungskompetenz des Nutzers
 - Handlungskompetenz: Mit dem erworbenen Wissen über das System kann der Nutzer seine Aufgaben erfüllen.
 2. Handlungsflexibilität:
 - System auch bei geänderter Aufgabenstellung effizient und effektiv nutzbar
 - Aufgaben können auf alternativen Wegen erledigt werden
 3. Aufgabenangemessenheit
 - Betrifft Durchführbarkeit, Effizienz und Qualität der Aufgabenbearbeitung
 - Kann ein Nutzer seine Ziele mit Hilfe der Anwendung erreichen oder muss er andere Systeme oder Medien zu Hilfe nehmen? (z.B. Mediensprünge)
 - Welcher Planungs- und Zeitaufwand wird benötigt?
 - Mit welcher Qualität kann das Ziel einer Aufgabe erreicht werden?

- **ISO/TS 16071: Design of accessible software**
 - Design auch für Menschen mit Einschränkungen der Sehfähigkeit, des Hörens oder von motorischen Fähigkeiten
 - Implizit auch für ältere Nutzer
 - Beispiele:
 - Einhaltung von Programmierstandards
 - Nutzung von multimedialen Informationsträgern

„Qual der Wahl“

- Für populäre Anwendungsbereiche (GUI, Web) gibt es zahlreiche verschiedene „Regelpakete“. Autoren verschiedener Pakete messen jedoch einzelnen Usability-Aspekten unterschiedliche Bedeutung zu.
 - Wahl eines Pakets ist z.T. Vertrauenssache, da man sich meistens nicht die Mühe macht, nachzulesen, auf welchen Erkenntnissen die einzelnen Regeln beruhen.
 - Eine eigene Zusammenstellung von Regeln unterschiedlicher Herkunft ist problematisch, da es zu Konsistenzproblemen / Widersprüchen kommen kann.
- Im Zweifelsfall auf standardisierte Regelpakete zurückgreifen.

„Nicht für alles gibt es Regeln“

- Regeln sagen nicht, wie man zu einem Entwurf kommt!
 - Regeln können helfen miteinander konkurrierende Entwürfe zu bewerten.
 - Kein Regelwerk ist vollständig.
- Ein Entwurf, der gemäß vorhandener Regeln schlecht abschneidet, kann trotzdem positive Qualitäten aufweisen, die von den Regeln nicht betrachtet werden. Und umgekehrt.

Interpretation erforderlich

- Designprinzipien sind meist sehr allgemein formuliert.
- Im konkreten Anwendungsfall ist stets eine Interpretation erforderlich, die mehr oder weniger fehleranfällig sein kann.

Dennoch

- Gute Kenntnis von Designprinzipien und Regeln hilft „Anfängerfehler“ zu vermeiden und fördert die kritische Auseinandersetzung mit Entwürfen.

Gesetze, Normen, Designprinzipien und Richtlinien

-

Grundlegende Designprinzipien

Prinzip P1: Einfacher und natürlicher Dialog

- Angemessenheit der Interaktion im Verhältnis zur Aufgabe
- Ablauf der Interaktion muss mit dem mentalen Modell übereinstimmen, das der Benutzer vom System gemäß dem konzeptuellen Modell aufbaut.
- Minimiere den Aufwand, den ein Benutzer betreiben muss, um Aufgaben im Anwendungsbereich auf Bedienaufgaben im Interface abzubilden.
- Präsentiere genau die zur Bewältigung der Aufgabe benötigte Information, nicht weniger aber auch nicht mehr.

■ Implikationen:

- präsentiere Information in natürlicher Bearbeitungsreihenfolge
- präsentiere keine unnötige Information und Optionen
- halte Navigation möglichst einfach
- keine unnötigen Fenster
- keine unnötigen Verschachtelungen

Beispiel:

- Gruppierung von GUI-Widgets in Fenstern und Layout-Boxen gemäß natürlicher Bearbeitungsreihenfolge

The screenshot displays the Deutsche Bahn (DB) ticket booking interface, organized into a sequence of steps: **Suche** (Search), **Auswahl** (Selection), **Ticket & Optionen** (Ticket & Options), **Zahlung** (Payment), **Prüfen & Buchen** (Check & Book), and **Bestätigung** (Confirmation). The **Suche** step is currently active.

Reiseauskunft (Travel Information):

- Strecke** (Route):
 - Start***: ☒ Bahnhof oder Haltestelle (Augsburg Hbf), ☐ Ort, Straße Hausnr., ☐ Sehenswürdigkeit (POI).
 - Ziel***: ☒ Bahnhof oder Haltestelle (München Hbf), ☐ Ort, Straße Hausnr., ☐ Sehenswürdigkeit (POI).
 - [-> Zwischenhalte angeben](#)
 - ☒ Einfache Fahrt, ☐ Hin- und Rückfahrt

Datum und Uhrzeit (Date and Time):

- Hinfahrt***: ☒ Abfahrt, ☐ Ankunft

Angaben zur Verbindung (Connection Details):

- Verkehrsmittel**: ☒ Alle, ☐ Alle ohne ICE, ☐ Nur Nahverkehr, [-> Erweiterte Verkehrsmittelwahl](#)
- Umsteigezeit**:
- ☒ Schnelle Verbindungen bevorzugen
- ☐ Fahrradmitnahme

Reisende (Travelers):

- Erwachsene**: **Kinder 6-14 J.**: **Kinder 0-5 J.**: [-> Mehr als 5 Reisende](#)
- 1 Erwachsener**:
- ☒ 2. Klasse reisen, ☐ 1. Klasse reisen

Reservierung (Reservation):

- ☒ Nur Sitzplatz, ☐ Nur Sitzplatzreservierung (keine Fahrkarte)

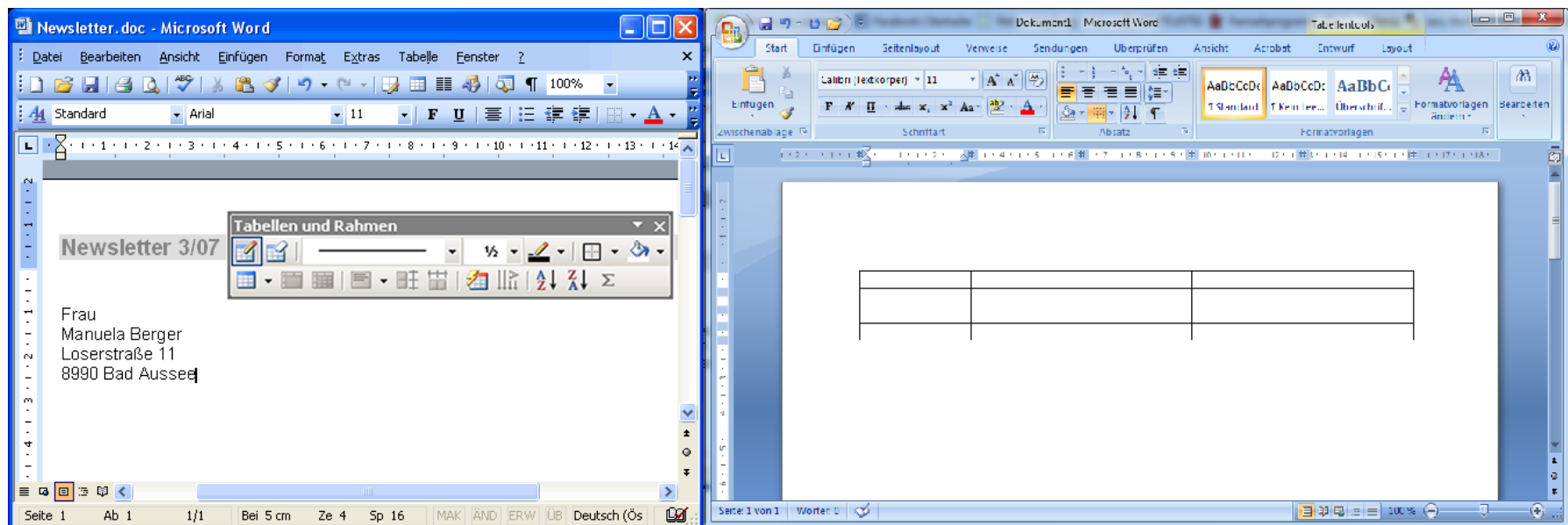
Buttons:

Häufige Fragen (Frequently Asked Questions):

- [-> Wo finde ich Tipps zur Fahrplanauskunft?](#)
- [-> Wo finde ich günstige Sparpreise?](#)
- [-> Wo finde ich Informationen zu Baustellen oder kurzfristigen Fahrplanänderungen?](#)
- [-> Alle häufig gestellten Fragen](#)

Verwendung des Prinzips:

- Effizienzvergleich unterschiedlicher Umsetzungen
- Beispiel: Anzahl der zu öffnenden / zu schließenden Fenster



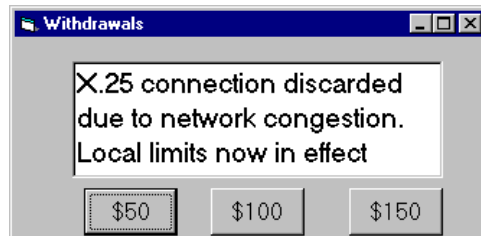
Prinzip P2: Verwende die Sprache der Nutzer

- verwende für Eingabeaufforderungen, Ergebnisausgaben, Hilfestellungen und für Status- und Fehlermeldungen die Terminologie der Anwendungsdomäne

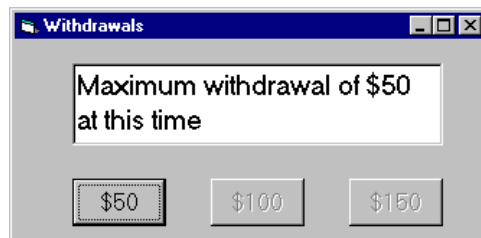
Implikationen:

- vermeide technischen Jargon in nicht technischen Anwendungen

schlecht:



besser:



Beispiel: Apple HCI-Guidelines für OS X

Table 3-3: Translating developer terms into user terms

Developer term	User term equivalent
Data browser	Scrolling list or multicolumn list
Dirty document	Document with unsaved changes
Focus ring	Highlighted area; area ready to accept user input
User-visible text	Onscreen text
Mouse-up event	Mouse click
Reboot	Restart
String length	Number of characters

Implikationen:

- Verwende mnemotechnisch sinnvolle Abkürzungen und selbsterklärende Icons

Beispiel: Save File: Ctrl + S
 Open File: Ctrl + O

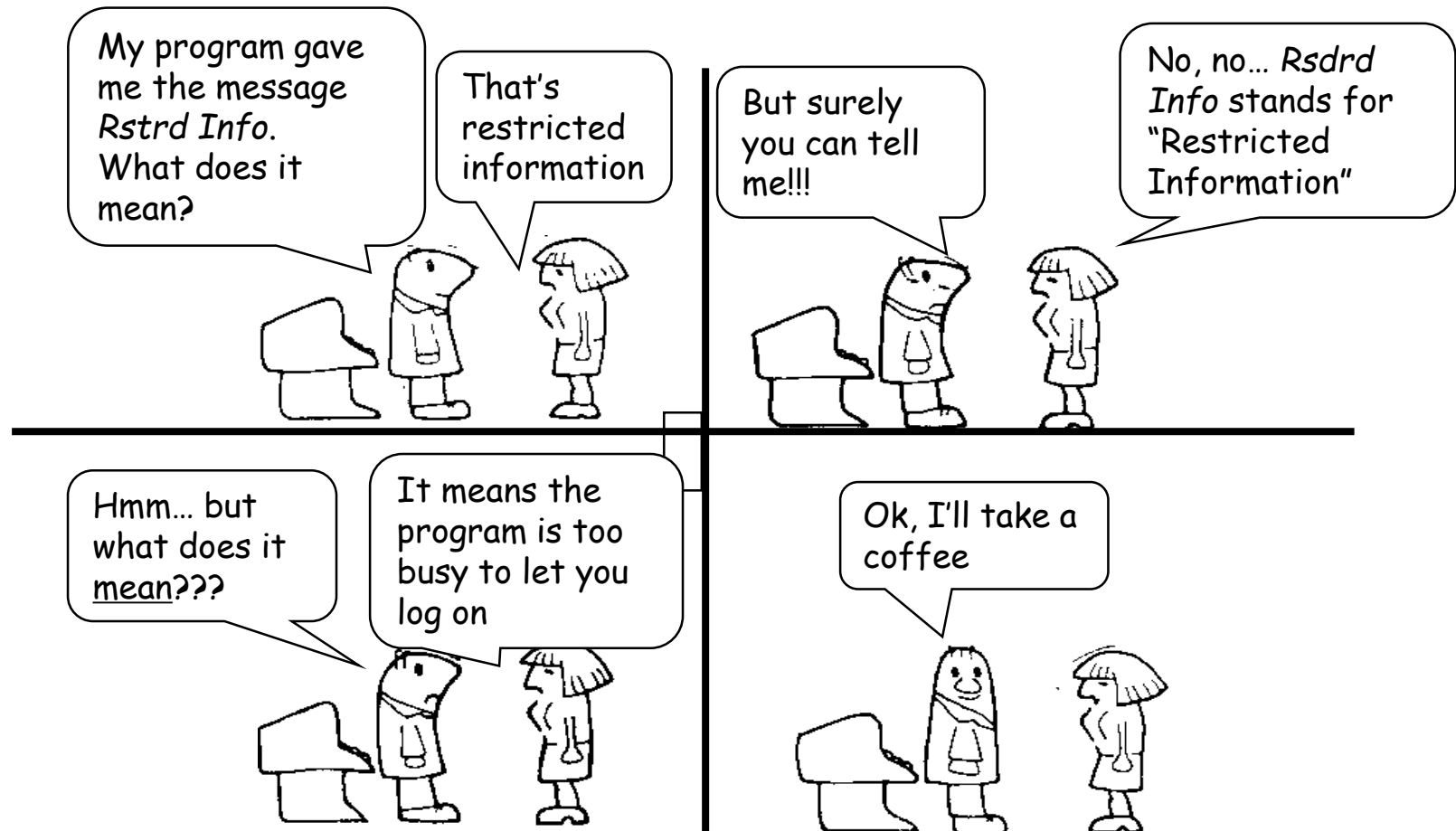


- Aber: Eignung für Nutzer anderer Sprachen?
- Sorge dafür, dass es für Aufgaben in der Domäne auch gleich lautende Aufgaben / Kommandos im Interface gibt.

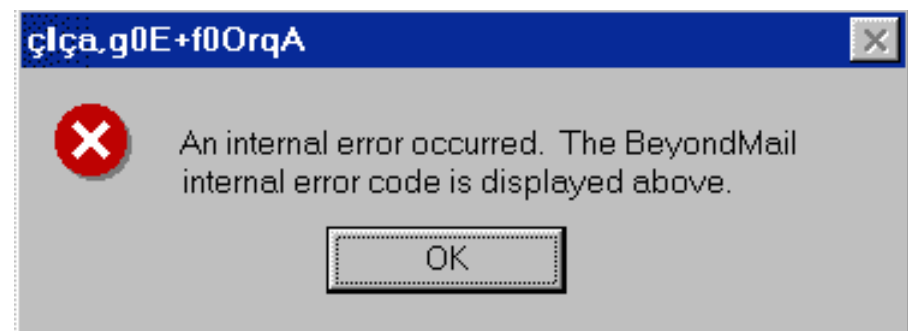
Beispiel:	DOS	UNIX
	move	mv
	Excel	GoogleDocs
	Summe	Sum

- **Implikationen:**

- vermeide kryptische Systemmeldungen (siehe auch Prinzip P9)



- Klassiker aus einer „Interface Hall of Shame“ zum Thema „verständliche Systemäußerungen“



<http://digilander.libero.it/chiediloapippo/Engineering/iarchitect/errormsg.htm>

Prinzip P3: Bewahre Konsistenz

- verwende gleiche Bedienabläufe für gleiche Aufgaben
- verwende konsistente Metaphern, Kodierungen (Namen von Befehlen, Abkürzungen, Farbkodierungen,...) und Stile (Schriftarten, Layout, ...)

Implikationen:

- vermeide unmotivierter Wechsel in:
 - Bedeutung von Icons
 - Layout und Anordnung von Schaltflächen
 - Form und Größe von Buttons
 - Farbgebung von Buttons
 - der Benennung von Kommandos:
 - Ja/Nein vs. Aktionsverben Accept/Dismiss
 - kein „Sprachgemisch“
 - Groß- oder Kleinschreibung
- **Konsistenz trägt zur Transparenz eines Systems bei !**



- **Implikationen:**

- Verwende einheitliche soziale Indikatoren beim Prompt-Design in Dialogsystemen

wish: eher formal

S: I am sorry. I didn't understand. Say the name of the account you wish to transfer money to. For example, you could say, „Savings account.“

could: eher informell

- **Implikationen:** Konsistenter Entwurf von Dialogstrategien
(Beispiel: Börsenkurse)

S: What would you like to do?

U: Play my watch list.

S: International Business Machines, eighty two, up one point three six.
Amazon, thirty six, down ...

U: **Next.**

S: Ford Motor Company, ten point ...

U: **Next.**

S: Charles Schwab Corporation, eleven ...

U: **Finished.**

S: What would you like to do next?

U: Tell me my open orders.

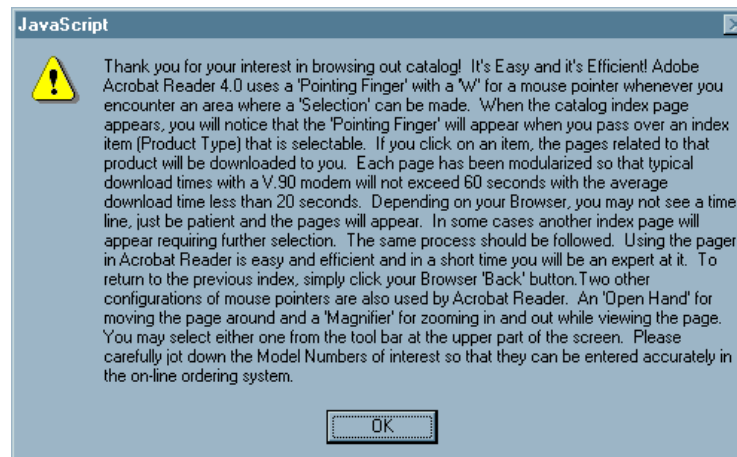
S: You have five open orders. Here's the first one: sell one hundred shares of Microsoft
...

U: **Next.**

S: Buy two hundred shares of Cisco Systems at seventeen ...

U: **Finished**

- **Prinzip P4: Entlaste den Nutzer von Gedächtnisaufgaben**
 - je weniger sich der Benutzer an Bedienabläufen merken muss, desto schneller erlernbar und erinnerbar ist das UI.
- **Implikationen:**
 - definiere möglichst wenige dafür generell einsetzbare Kommandos (z.B. keine speziellen Speicherbefehle für verschiedene Datentypen)
 - verwende kurze informative Systemausgaben



Aus der Gedächtnisforschung ist bekannt:

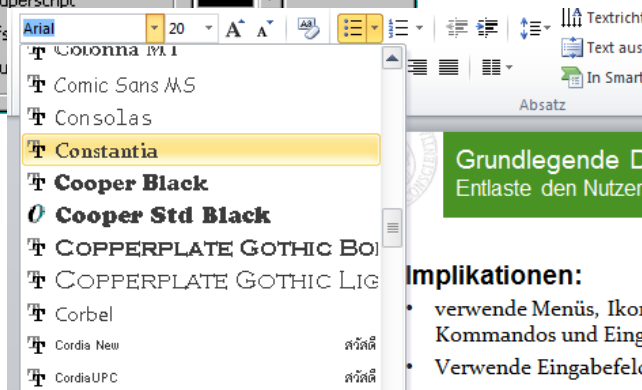
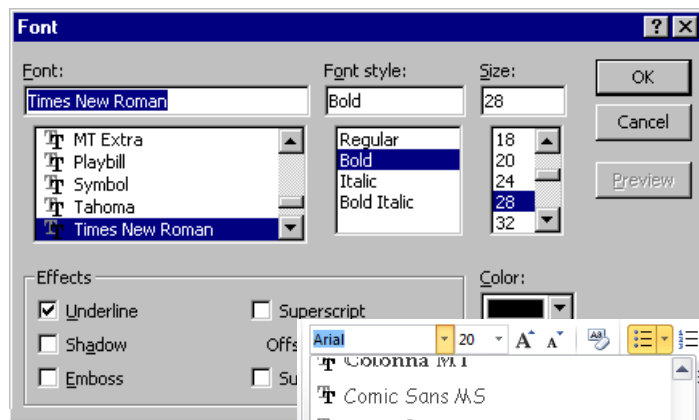
- Unterscheidung zwischen Kurz- und Langzeitgedächtnis
- Selektive Arbeitsweise des menschlichen Gedächtnis. Nur ein Bruchteil dessen, was wahrgenommen wird, findet Einzug in das Gedächtnis.
- Episoden werden besser erinnert als einzelne Fakten.
- Der Kontext spielt eine wichtige Rolle, ob z.B. Kommandos und Arbeitsabläufe behalten werden.

Aus der Gedächtnisforschung ist bekannt:

- Menschen sind im Allgemeinen besser:
 - im Wiedererkennen als im Erinnern von Dingen
 - im Erinnern von Bildern als im Erinnern von Namen/Befehlen
- Hierin begründet sich ein wesentlicher Vorteil von GUIs gegenüber Kommandozeilen Interfaces:
 - GUI ↔ Wiedererkennen von Icons / Bildern
 - Kommandozeilen UI ↔ Erinnern von Befehlen

Implikationen:

- verwende Menüs, Ikonen, Auswahlboxen anstelle von Kommandos und Eingabefeldern
- Verwende Eingabefelder mit Format-Constraints




Implikationen:

- verwende Menüs, Ikonen, Kommandos und Eingabefelder
- Verwende Eingabefelder

Theorie von George Miller:

- Wegen der beschränkten Kapazität des menschlichen Kurzzeitgedächtnisses liegt die Anzahl der bei einmaliger Präsentation erinnerbaren Objekte bei 7 ± 2
- **Aber:** Es gibt viele Beispiele für eine falsch verstandene Anwendung dieser Regel!
 - max. 7 Einträge pro Menü
 - max. 7 Bullet Points in Liste
 - max. 7 Icons in Toolbar
 - ...
- Menschen können beliebig lange Listen sequentiell durchsuchen, ohne sich die Einträge merken zu müssen!

Beispiel: Börse

S: Welcome to Princeton Brokerage. I can help you get quotes, set up a watch list, trade equities or options, and access your account information.
To get a quote, simply say the name of the company.
To set up a watch list, say, „Set up watch list,“ and then you will be asked for each company you want to put in your list. After you set up a watch list, you can get quotes ...



S: Welcome to Princeton Brokerage. I can help you get quotes, set up a watch list, trade equities or options, and access your account information.
What would you like to do?
U: I'd like a quote.
S: To get a quote, just say the name of the stock, mutual fund, or index name.
U: IBM
S: International Business Machines is selling at eighty two, up one point three six.
U: Cisco.
S: ...



- **Prinzip P5:** Lenke die Aufmerksamkeit auf wichtige Information

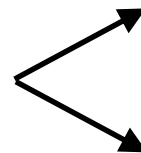
- verwende geeignete gestalterische Mittel, um wichtige Information gegenüber weniger wichtiger Information hervorzuheben.

- Implikationen:

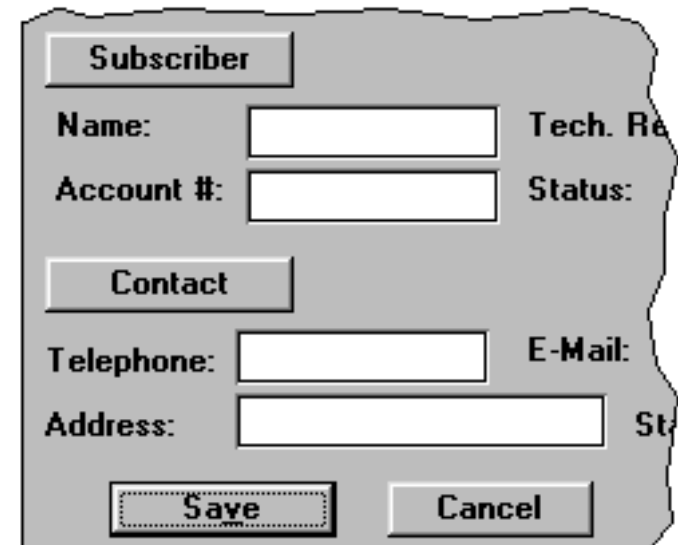
- hebe wichtige Information visuell hervor
- vermeide jedoch unmotivierte Hervorhebung



Überschrift, kein Button
Hervorhebung provoziert
Fehlbedienung




Button



Beispiel: Flughafen-Information


S: United Airlines flight 47 leaves New York Kennedy Airport at 1 pm from gate 36 and arrives at Boston Logan at 1:45 pm at gate 22.
American Airlines flight 243 leaves New York Kennedy Airport at 2:15 pm from gate 12 and arrives at Boston Logan at 3pm at gate 47.
American Airlines flight ... Which one would you like?



S: There are flights at 1pm, 2:15pm, 3:45pm and 5pm. Which one would you like?

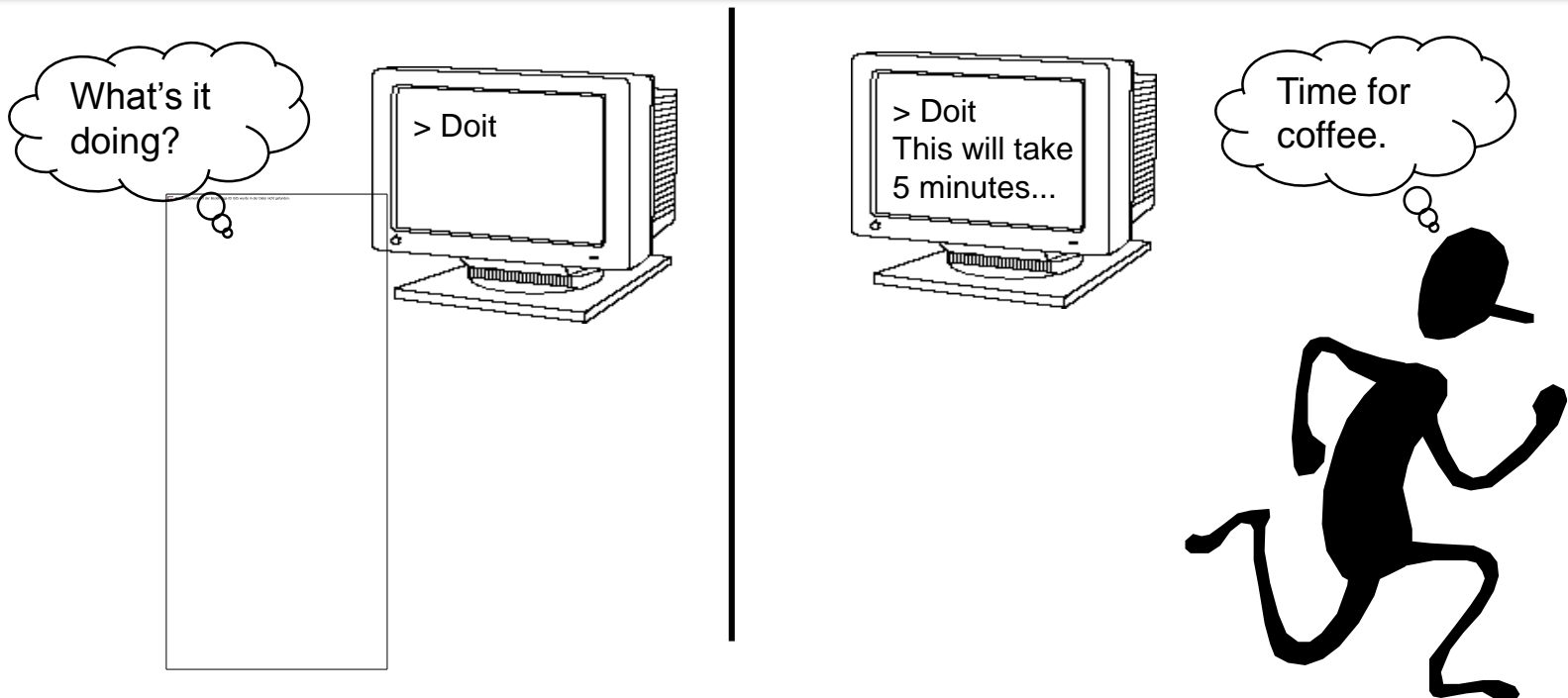
U: How about the 2:15 flight?

S: American Airlines flight 243 leaves New York Kennedy Airport at 2:15 pm and arrives at Boston Logan at 3pm. Would you like to book this flight?



• Prinzip P6: Gib sinnvolle Rückmeldungen

- quittiere Eingaben, am Besten durch Rückmeldung, wie sie vom System interpretiert werden
- Nutzer sollte stets wissen, was das System in Bezug auf die Aufgabe gerade tut

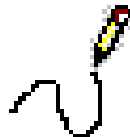


Implikationen:

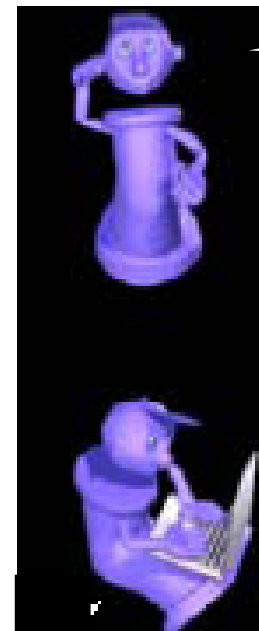
- kontinuierliche Rückmeldung über eingestellten Verarbeitungsmodus
- Beispiele:
 - Wechsel des Cursors



aktiverModus



- Indikation des Systemzustands durch Mimik und Gesten im Sprachdialoginterface Smartkom (<http://www.smartkom.org/>)



Eingabe nicht
verstanden

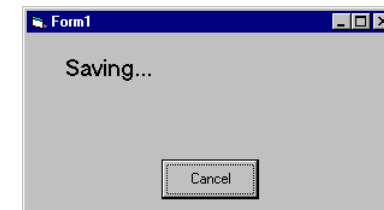
Auftrag wird
bearbeitet

Implikationen:

- Rückmeldung möglichst spezifisch machen

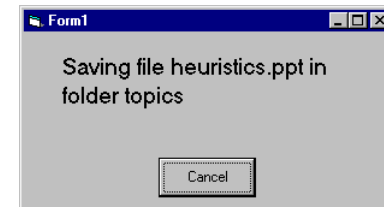
Beispiel: Speichern einer Datei

Variante 1: Was wird gespeichert?



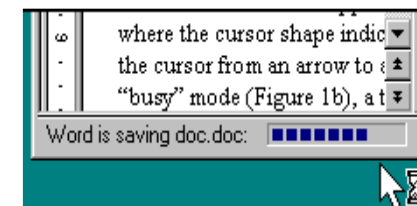
besser: Variante 2:

Zu speichernde Datei wird angezeigt



noch besser: Variante 3:

Speichervorgang wird im Kontext
angezeigt

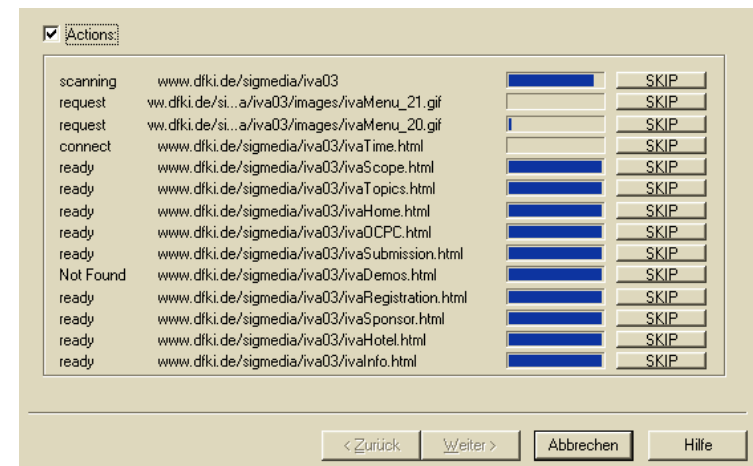


Beispiele:

Rückmeldung einer Aufgabensequenz:
Rückmeldung der jeweils aktiven Aktion
(z.B. mehrere Files kopieren)

Transfer		Queue				
Source File	S.	D..	Size	Status	Speed	
↑ img00111.gif	C..	/...	4,298	Complete	3,9 KB/s	
↑ img00112.gif	C..	/...	2,992	Complete	3,5 KB/s	
↑ img00113.gif	C..	/...	5,732	Complete	6,0 KB/s	
↑ img00114.gif	C..	/...	1,106	Complete	1,5 KB/s	
↑ img00115.gif	C..	/...	2,194	Complete	2,9 KB/s	
↑ dvs-01-intr...	C..	/...	4,211...	84%	14,4 KB/s	
↑ dvs-02-mg...	C..	/...	167,936	Queued	0,0 KB/s	
↑ dvs-03-cod...	C..	/...	596,992	Queued	0,0 KB/s	
↑ dvs-04-zahl...	C..	/...	561,152	Queued	0,0 KB/s	
↑ dvs-05-logi...	C..	/...	1,659...	Queued	0,0 KB/s	
↑ dvs-06-sch...	C..	/...	474,624	Queued	0,0 KB/s	
↑ dvs-07-swe...	C..	/...	616,960	Queued	0,0 KB/s	

Rückmeldung nebenläufiger Aufgaben:
multiple Progress-Bars
(z.B. Website kopieren mit mehreren
Threads)



Implikationen:

- Es sollte so wenig Zeit wie möglich zwischen Eingabe und Rückmeldung vergehen

Beispiel: Interpretation zeitverzögerter Rückmeldungen

Verzögerung

Interpretation

bis maximal 0.15 Millisekunden

direkt / unverzögert

1 Sek

Verzögerung wird wahrgenommen, oft noch nicht als Unterbrechung empfunden (Ausnahme: Schreiben)

10 Sek

Grenze, ab der die Aufmerksamkeit des Nutzers auf Dialog verloren geht

mehr als 10 Sek

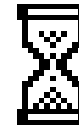
Nutzer versucht, Wartezeit anderwärtig zu nutzen. Ablenkung vom Dialog

Implikationen:

- Verwende spezielle Indikatoren zur Anzeige der Wartezeiten

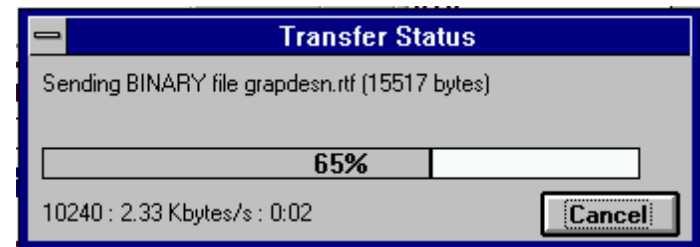
kurze Wartezeiten:

Ändern des Cursors / animierte Ikonen



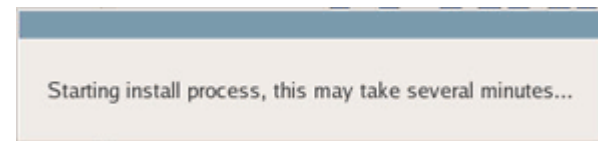
längere Wartezeiten:

Verwendung von „Progress-Bars“



nicht-vorhersagbare Wartezeiten:

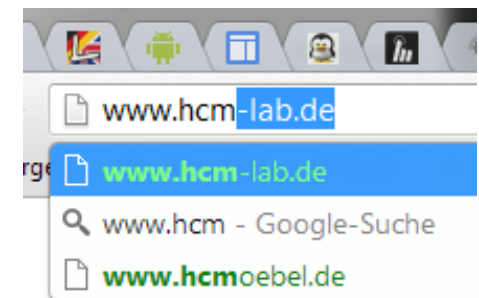
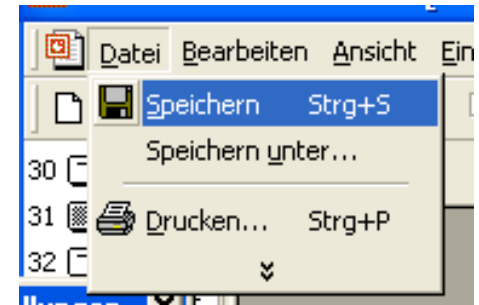
Angabe einer Schätzung [Min, Max]



- **Prinzip P7:** Biete dem geübten Nutzer Abkürzungen an

Implikationen:

- biete Möglichkeiten zur Beschleunigung von Eingaben
 - Funktionstasten mit Icons für Kommandos
 - Shortcuts (z.B. Ctrl+, Alt+, ...)
 - Erstellung von Makros
 - Kontext-Menüs (d.h. Häufigkeits-gesteuerte Umsortierung der Optionen)
 - automatische Eingabe-Vervollständigung durch Puffern von System-Eingaben (Kommandos, URLs, ...)

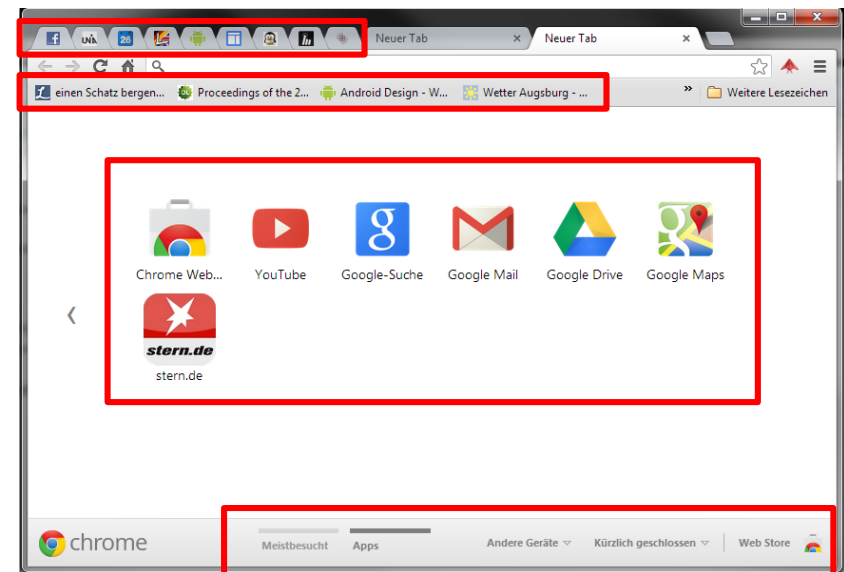


Beispiele: Telefondialogsysteme

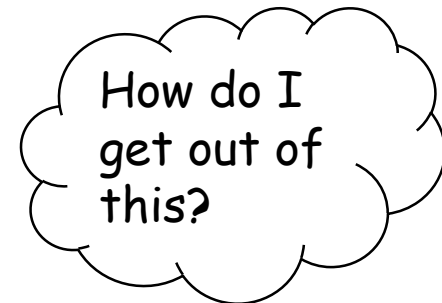
- Ansagen können vom geübten Nutzer „übersprochen“ werden
- System akzeptiert wahlweise sprachliche Eingaben oder Zifferneingaben

Weitere typische Shortcuts in einem GUI

- Anpassbare Toolbars
- Historie der zuletzt bearbeiteten bzw. betrachteten Dokumente oder Webseiten
- Menü durch Doppelklick auf Objekte
- Sprung an Anfang/Ende

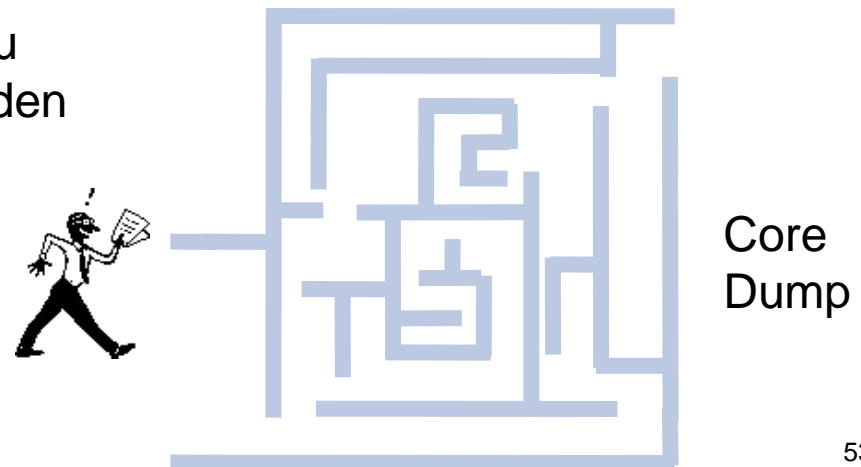


- **Prinzip P8:** Markiere Ausgänge deutlich
 - vermeide, dass sich der Nutzer „gefangen“ fühlt



Implikationen:

- Soweit möglich, alle Arbeitsabläufe und Aktionen mit Ausgängen versehen (z.B. Abbruch- oder Exit-Optionen)
- Unterbrechung mit Wiederaufsetzmöglichkeit (Interrupt-Resume)
 - um andere Aufgabe während einer Session auszuführen
 - um Arbeit zu späterem Zeitpunkt wieder aufzunehmen
 - universelles Undo für Rückkehr zu vorangegangenen Systemzuständen



- **Prinzip P9:** minimiere die Gefahr einer Fehlbedienung
 - verhindere Fehlbedienungen so weit möglich
 - erkenne aufgetretene Fehler und verstehe die Ursache
 - warne den Nutzer, wenn ungewöhnliche Situation eingetreten ist

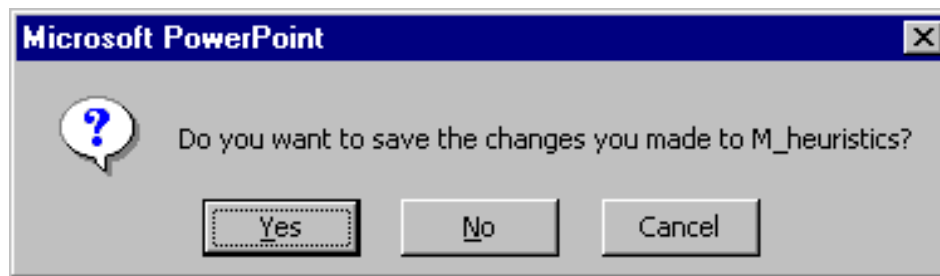
Fehlerarten

- Einteilung nach **Intention**:
 - Versehen / Ausrutscher (versehentliche Bedienfehler):
 - richtige Intention aber falsche Ausführung
 - Ursachen: mangelndes Training, Unaufmerksamkeit, Nachlässigkeit (Überschätzung) auf bekanntem Terrain,...
 - Missverständnisse (echte Bedienfehler):
 - falsche Intention
 - falsches Verständnis vom System (falsches mentales Modell)
- Einteilung nach **Entstehung** in verschiedenen Dialogphasen:
 - Lesen, Erkennen → Perzeptionsfehler
 - Denken → kognitive Fehler
 - Antworten, Agieren → motorische Fehler

- Unterbewusste Fehler, begünstigt durch Ablenkungen aller Art
- Treten oft in Tätigkeiten auf, die eingeübt sind oder sogar automatisch ablaufen.

Beispiele:

- E-Mail wird versehentlich ohne Attachment abgeschickt
- Statt Cancel wird Yes geklickt



Arten, Ursachen und Abhilfen:

- **Beschreibungsfehler:** Aktionen sind ähnlich und können verwechselt werden (z.B. ähnlich aussehende Icons, Buttons, Schalter usw.)
 - **Abhilfen:**
 - Verwendung hinreichend unterscheidbarer Ikonen
 - „Rückgängig machen“ ermöglichen
 - Bestätigung erfragen

Arten, Ursachen und Abhilfen:

- **Datenbedingte Fehler** durch ähnliche oder interferierende Daten
 - Beispiele: Telefonnummern, Namen (z.B. Katrin, Karin,...)
- **Abhilfen:**
 - Formateinschränkungen in Eingabefeldern
 - Plausibilitätsüberprüfung von Eingaben
 - Eingaben sichtbar machen und „stehen“ lassen (Feedback)
 - Präsentation der Ergebnisse zu Eingaben in Bezug setzen

Arten, Ursachen und Abhilfen:

- **Faden verloren:** Teile einer Handlung werden vergessen (Warum habe ich das jetzt gemacht? Wo war ich stehengeblieben? usw.)

→ Abhilfen:

- übergeordnetes Ziel und Aufgabenkontext „sichtbar“ machen
- Position im Workflow anzeigen (oder wenigstens in Historie nachschlagbar machen)



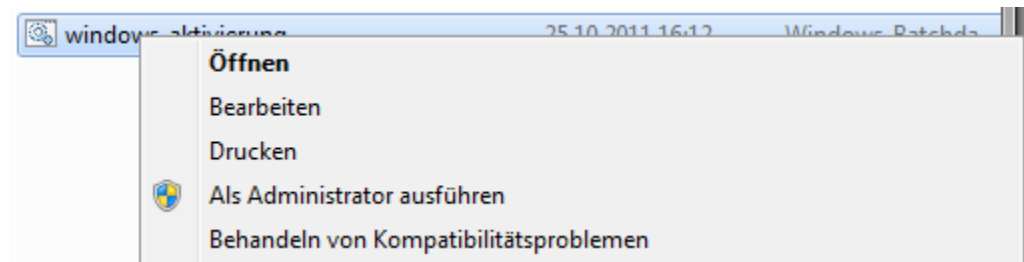
Arten, Ursachen und Abhilfen:

- **Modusfehler:**

- falsche Aktion in falschem Kontext
- Aktion wäre korrekt in einem anderen Kontext

→ **Abhilfen:**

- möglichst wenig Modi benutzen
- Notwendigkeit häufiger Modi-Wechsel minimieren
- aktuellen Modus gut sichtbar machen (multiple Indikatoren)

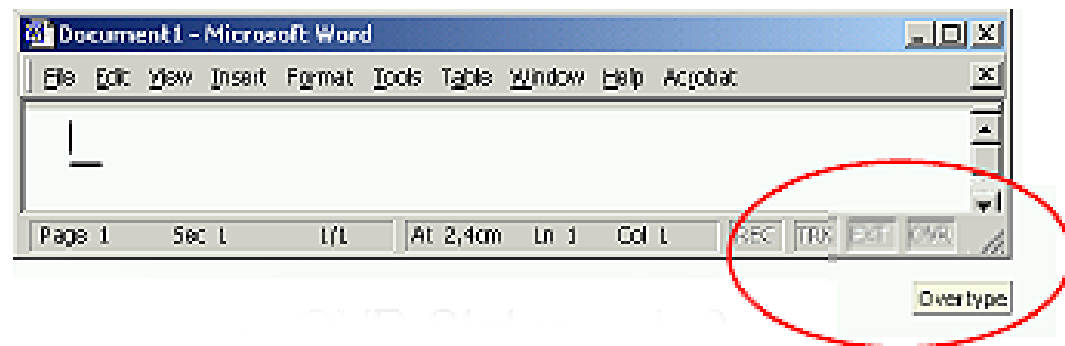


Ursachen:

- fehlende, falsche oder unzureichende perzeptorische Hinweise
- unterschiedliche Objekte sind wegen zu ähnlicher Darstellung nicht voneinander unterscheidbar
- Aufmerksamkeit wird nicht hinreichend geweckt (vgl. Prinzip P5)
- zu kleine bzw. ungünstig platzierte Statushinweise
- zu kleine Schrift, schlechte Farbkontraste, usw.
- nicht ausreichendes Feedback (z.B. zu kurze Dauer)

Beispiel:

Anzeige des Status
(OverType) wird leicht
übersehen



Ursachen:

- Hohe Belastung des Gedächtnisses oder des Denk-/ Abstraktionsvermögens
- Falsches mentales Modell vom System
- Unzureichende Hilfestellung vom System
- Fehlende Kontext- oder Statusinformation

Merkmale:

- werden oft schwer als solche erkannt
- Fehler werden oft einfach “wegerklärt”
- sozialer Druck kann Fehler hervorrufen (→ Angst zu fragen)

Beispiel:

1. Neuen Brief schreiben mit Vorlage eines alten Briefs
 2. Beim Speichern Umbenennen vergessen
- aus Versehen wird alter Brief gelöscht

Ursachen:

- Belastung von Augen-Hand-Koordination
- Voraussetzung spezieller motorischer Fähigkeiten (z.B. Tippen)
- Zeitdruck
- Ungünstige Bewegungen (z.B. AltGr-@)
- Sehr ähnliche Bewegungen
 - Beispiel: Doppelklick vs. Markieren und Klicken
- Ungeeignete Eingabegeräte
 - zu kleine Handy-Tastaturen
 - manche Laptop-Tastaturen



versehentliches Berühren des Touchpads mit Handballen beim Tippen verschiebt den Cursor!

Abhilfen:

- Abschalten von Touchpads ermöglichen
- Tastensperre
- Auto-Korrektur



Beispiel: Sprachdialogsysteme

- **Ziel:**
 - Vermeidung von Erkennungs- und Benutzerfehlern durch geeignetes Dialogdesign
- **Design der Benutzerführung:**
 - Informative, prägnante Systemäußerungen
 - Anbieten von Hilfe, insbesondere wenn Benutzer in Schwierigkeiten gerät
 - Implizite oder explizite Verifikation
 - Einprägsamer, intuitiver Dialogfluss
- **Design der Spracherkennung:**
 - Bereitstellen der häufigsten Synonyme und alternativen Phrasen
 - Eventuell Einfügen geeigneter Füllwörter („bitte“, „ich möchte“ etc.)

- **Prinzip P10: konstruktive Behandlung von Fehlern**
 - Informiere sinnvoll über fehlgeschlagene Kommandos und mögliche Alternativen.
 - Ermögliche die Rücknahme von Aktionen / Kommandos

Beispiel für fragwürdige Fehlermeldungen:



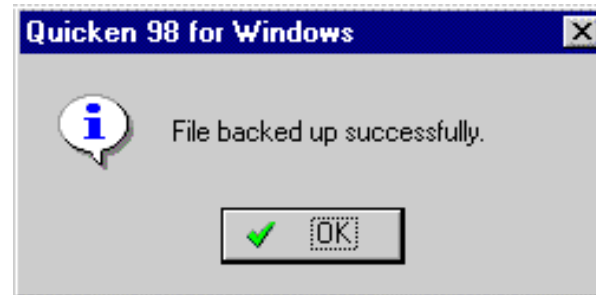
verständlich?
(was ging daneben)

hilfreich?
(was soll ich tun)

witzig?

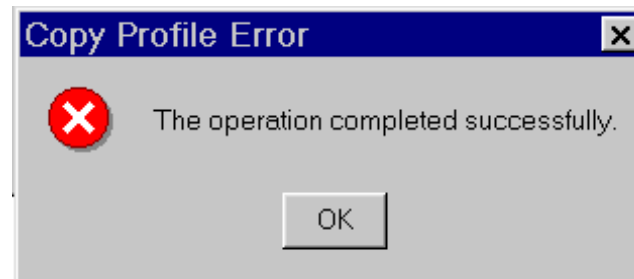
Beispiele für fragwürdige Fehlermeldungen:

Ist die erfolgreiche
Ausführung eines
Kommandos was
Besonderes?



Quicken 98

Fehler oder keiner?



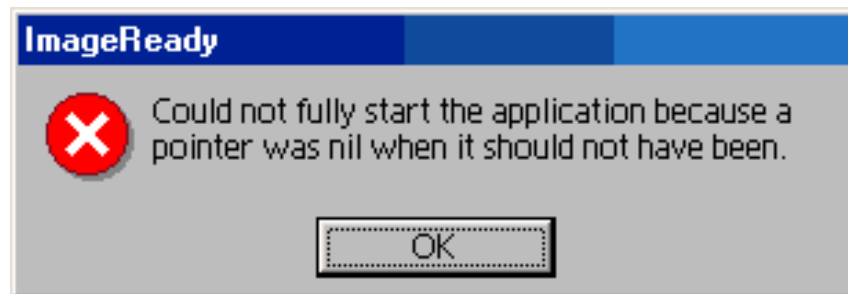
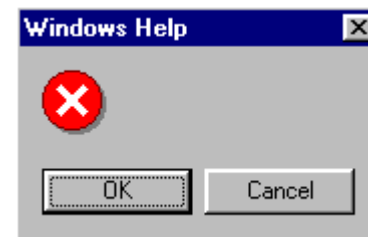
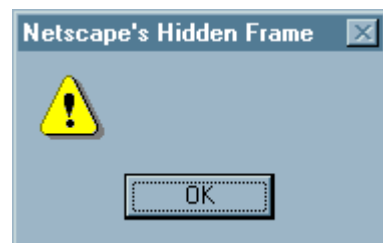
*Microsoft's NT
Operating System*

ohne Worte

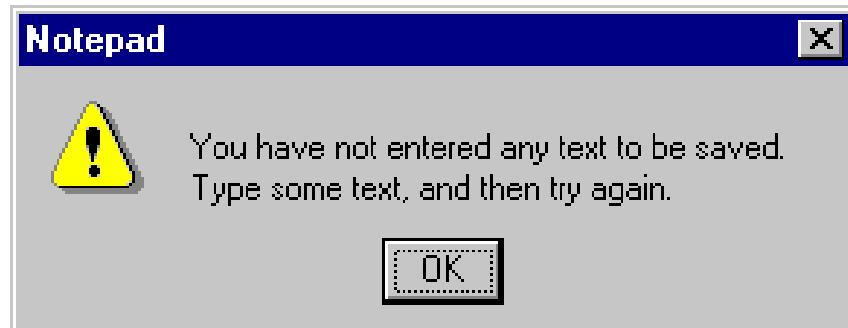


Mac OS

Beispiel für fragwürdige Fehlermeldungen:



Adobe's
ImageReady



*Windows
Notepad*

weitere Beispiele: <http://digilander.libero.it/chiediloapippo/Engineering/iarchitect/errmsg.htm>

Implikationen aus Prinzip P10:

- Fehlermeldungen sollen informativ, konstruktiv und freundlich sein!



Try again, bonehead!



Error 25



Cannot open this document



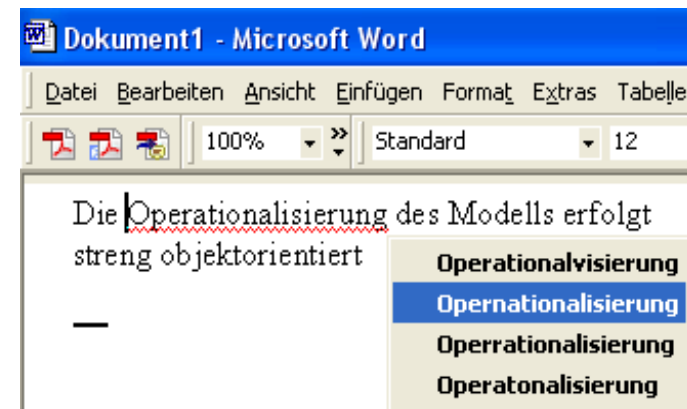
Cannot open “chapter 5” because the application “Microsoft Word” is not on your system



Cannot open “chapter 5” because the application “Microsoft Word” is not on your system. Open it with “StarOffice” instead?

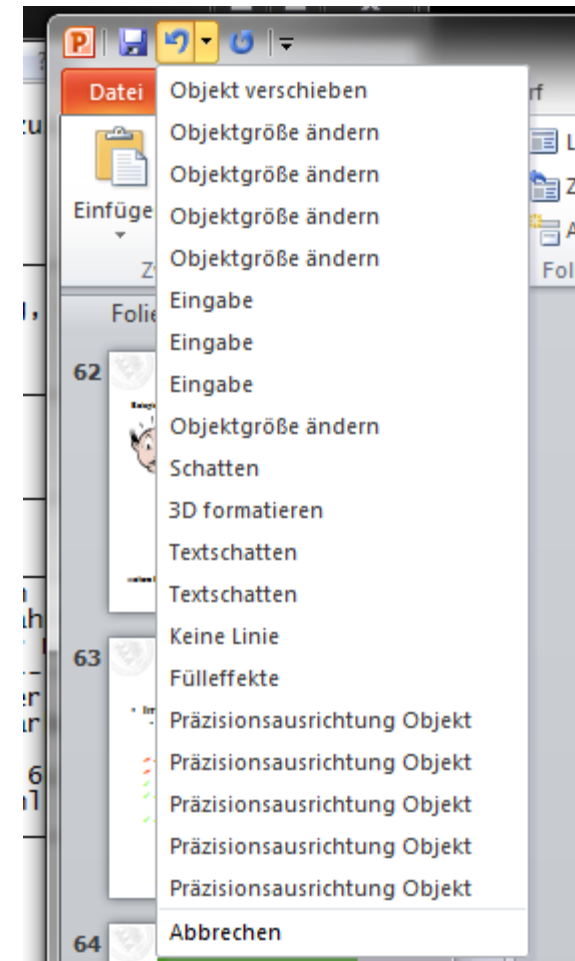
Implikationen aus Prinzip P10:

- manche Systeme können Fehler erkennen und machen konstruktive Korrekturvorschläge:
- manche Systeme vermuten jedoch Fehler, wo keine sind:



Implikationen aus Prinzip P10:

- so weit machbar, sollte dem Nutzer stets die Möglichkeit eingeräumt werden, abgesetzte Kommandos / Aktionen rückgängig zu machen.
- Gute Systeme haben einen ausgereiften Historie-Mechanismus, der die Annullierung von Befehlssequenzen ermöglicht.



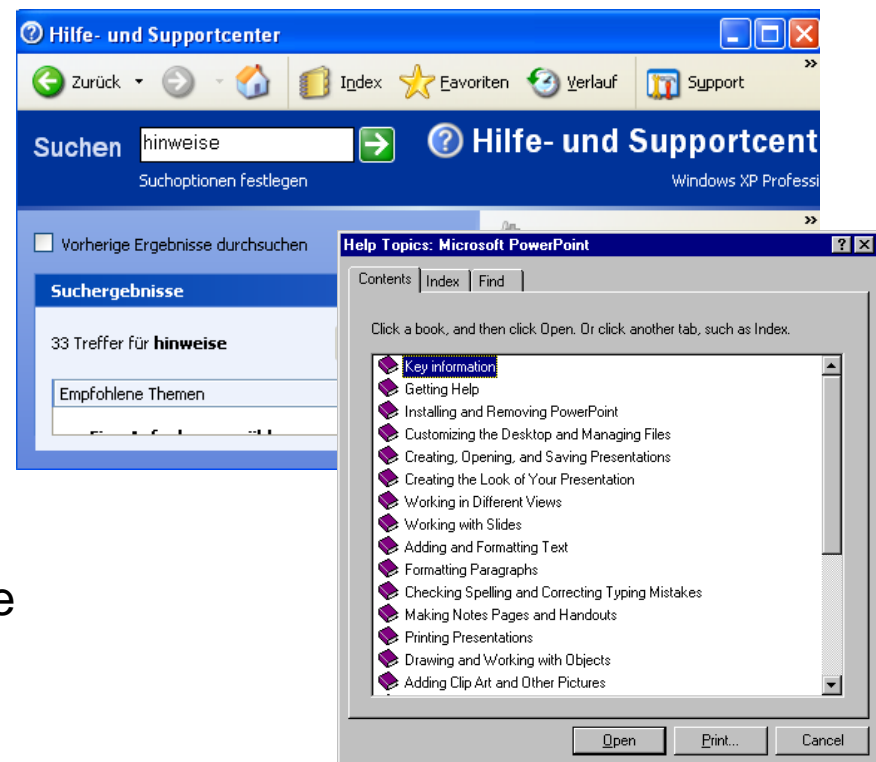
Beispiele für Fehlermeldungen in Dialogsystemen:

Schlecht	Besser
Das Telefon ist nicht verfügbar	Bitte legen Sie eine SIM Karte ein
Ich habe Sie nicht verstanden	Ich konnte Sie nicht verstehen, bitte sprechen Sie etwas lauter.
Ich habe Sie nicht verstanden	Bitte sprechen Sie die Telefonnummer als eine Folge von Ziffern oder sagen Sie Abbruch.

- **Prinzip 11: leiste adäquate Hilfestellung**
- Nutzer muss jederzeit Zugang zu Hilfestellung haben
- Art der Hilfe muss situations- und zielgruppenspezifisch sein

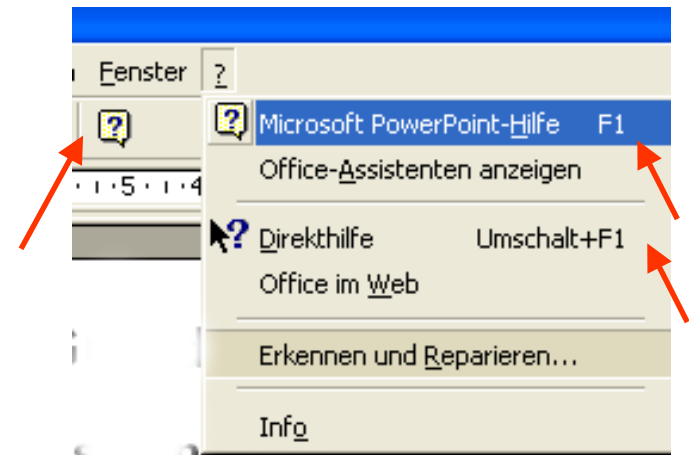
Arten der Hilfe:

- Hilfe für Erstnutzer
 - Einstiegstutorien
 - geführte Touren
 - Handbuch
- Online-Hilfe
 - Thesauri / Hypertext
 - Erinnerungen, Tipps
 - kontext-sensitive Hilfe
 - adaptive (intelligente) Hilfe
 - ...



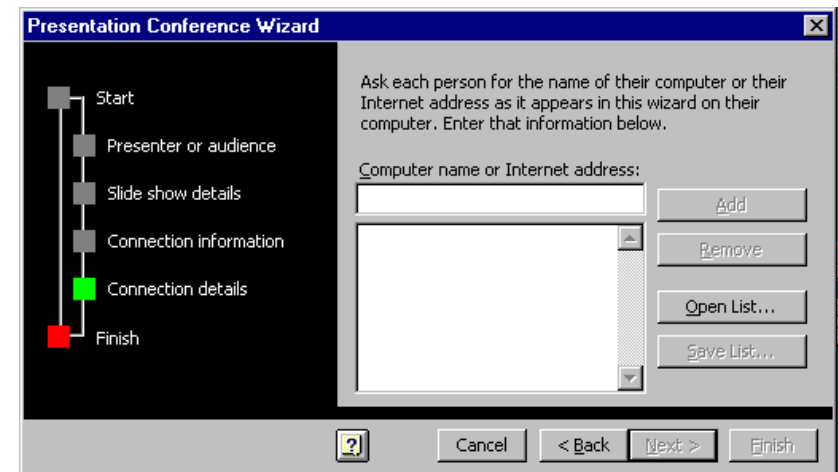
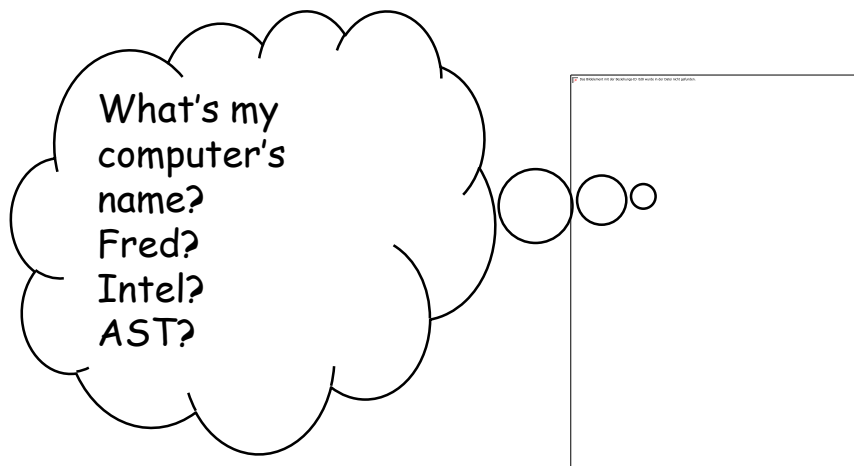
Implikationen aus Prinzip P11:

- einheitlicher Zugang zu Hilfestellungen
 - F1
 - ? Ikon
 - „Help“-Kommando



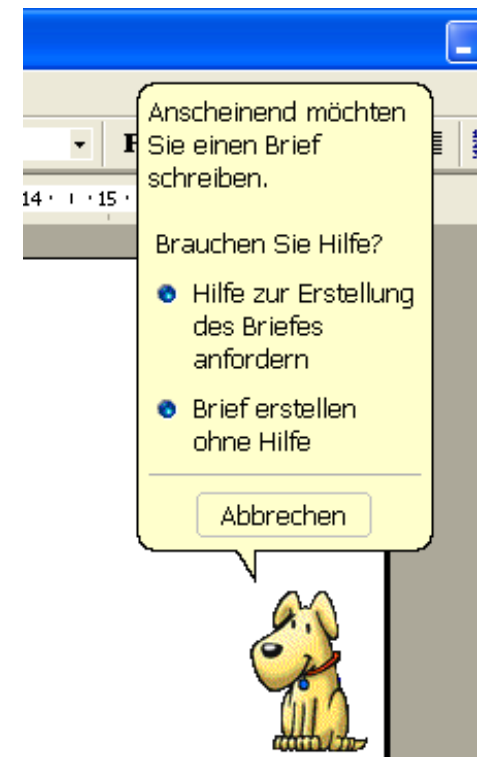
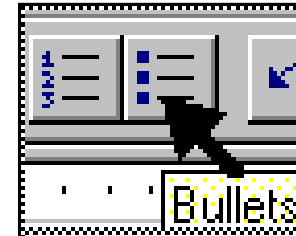
Spezielle Hilfeformen:

- Wizards / Assistenten
 - Skript führt Nutzer durch Anwendung
 - Gefahr stecken zu bleiben



Spezielle Hilfeformen:

- Hinweise / Tooltips
 - einfacher Abruf durch „Mouse over“
- Aktive Hinweise:
 - System wertet Benutzereingaben aus
 - Stellt Hypothese über das vermeintlich verfolgte Ziel auf
 - Falls es der Ansicht ist, dass es eine bessere Möglichkeit gibt, das Ziel zu erreichen, gibt es unaufgefordert einen Hinweis.
- Aber:
 - Intentionserkennung ist meist schwierig
 - Aktive Hilfe ist nicht immer willkommen



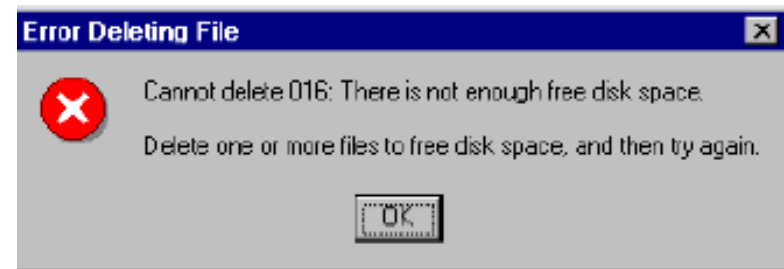
Implikationen aus P11:

- nur sinnvolle Hinweise auf durchführbare Aktionen geben!



Hilfe im Fehlerfall:

- Hilfestellung bei der Lokalisierung des Fehlers (z.B.: indem Cursor auf Fehlerfeld positioniert wird)
- Intelligent helfen. Nutzer erwartet Antwort auf Fragen:
 - Warum trat der Fehler auf?
 - Wie lässt er sich beheben?
 - Wie kann erneutes Auftreten verhindert werden?



Pakete von Designprinzipien und Richtlinien

1. **Strive for consistency** (vgl. Prinzip P3, P2)
2. **Enable frequent users to use shortcuts** (vgl. P7)
3. **Offer informative feedback** (vgl. P6)
4. **Design dialog to yield closure** (vgl. P1, P2)
5. **Offer simple error handling** (vgl. P10)
6. **Permit easy reversal of actions** (vgl. P10)
7. **Support internal locus of control**
8. **Reduce short-term memory load** (vgl. P4)

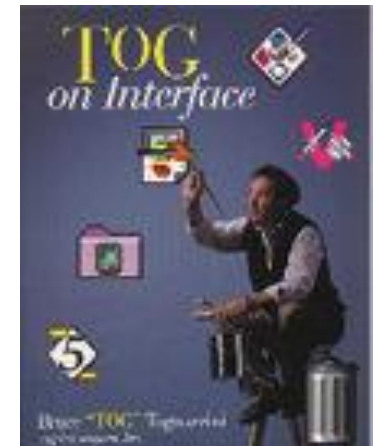


- 1. Visibility of system status (vgl. P6)**
- 2. Match between system and the real world (vgl. P2)**
- 3. User control and freedom**
- 4. Consistency and standards (vgl. P3, P5)**
- 5. Error prevention (vgl. P9, P10)**
- 6. Recognition rather than recall (vgl. P4)**
- 7. Flexibility and efficiency of use (vgl. P7)**
- 8. Aesthetic and minimalist design (vgl. P1)**
- 9. Help users recognize, diagnose, and recover from errors (vgl. P10)**
- 10. Help and documentation (vgl. P11)**



Bruce Tognazzini's First Principles of Interaction Design

- Antizipation
- Autonomie
- Farbenblindheit
- Konsistenz
- Defaultwerte
- Effizienz des Nutzers
- Erforschbare Oberflächen
- Fitts' Gesetz
- Oberflächenelemente
- Wartezeitminimierung
- Erlernbarkeit
- Verwendung von Metaphern
- Schutz der Arbeit des Nutzers
- Lesbarkeit
- Zustandsanzeige
- Sichtbare Navigation



- Nutzerzentriertes Design durchführen
- Kenne den Nutzer
- Involviere den Nutzer
- Verhindere Fehler durch den Nutzer
- Optimierte die Nutzerinteraktionen
- Lass die Kontrolle beim Nutzer
- Hilf dem Nutzer
- Gib ein aufgabenbasiertes mentales Modell vor
- Sei konsistent
- Halte es einfach
- Nutze ein kognitiv wenig belastendes Design
- Nutze Analogien aus der realen Welt

- Gib ein informatives sofortiges Feedback
- Zeige den Systemstatus an
- Nutze nutzerzentrierte Sprache
- Biete konstruktive Unterstützung an
- Blamiere den Nutzer nicht
- Vermenschliche nicht zu sehr
- Nutzeraktionen sollen reversible sein
- Hole dir Aufmerksamkeit in vernünftigen Masse
- Organisiere die Bildschirme, um die Komplexität zu reduzieren
- Beachte individuelle Unterschiede

Ansatz: Designmethode + 5 Guidelines

Designmethode:

- Schritt 1: Focus each screen on a single task.
- Schritt 2: State the task.
- Schritt 3: Make the screen's contents suit the task.
- Schritt 4: Offer links to secondary tasks.

Design-Regeln

- R1: Use consistent screen templates.
- R2: Provide screens for starting tasks.
- R3: Make it obvious how to carry out the task with the controls on the screen.
- R4: Provide an easy way to complete a task and start a new one.
- R5: Make the next navigational step obvious.

- Der Satz an grundlegenden / allgemeinen Design-Prinzipien variiert je nach Quelle:
 - Ben Shneiderman's „8 Golden Design Rules“
 - Jakob Nielsen's „10 Usability Heuristics“
 - Bruce Tognazzini's „First Principles of Interaction Design“
 - ...
- Trotz breiter Überdeckung werden bestimmte Aspekte (Usability-Qualitäten) anders gewichtet.