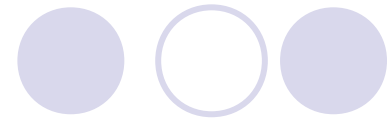
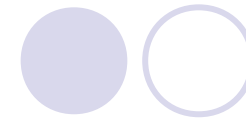


UI Design für Einsteiger

Erstellung effektiver

Anwendungsoberflächen

Heutige Anwendungen



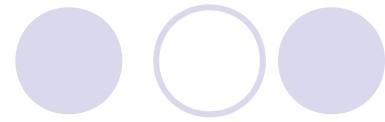
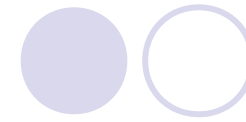
- ◆ **Anwendungen definieren sich durch eine abgegrenzte Funktionalität**
- ◆ **Diese Funktionalität steht dem Anwender zur Verfügung**
- ◆ **Der Anwender „nutzt“ die Funktionalität über die Anwendungsoberfläche oder das User Interface, kurz UI**

Problem



- ◆ In die Anwendungen wird viel Zeit investiert
- ◆ Dennoch ist das UI oft das Letzte, worum man sich kümmert
- ◆ Wenn man sich überhaupt darum kümmert
- ◆ Das User Interface ist *eigentlich* das Wichtigste, wird aber oft am meisten vernachlässigt

Die 5 UI-Sünden



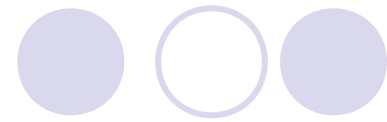
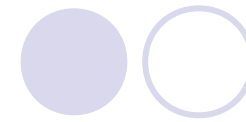
- ◆ Überfrachtetes UI
- ◆ Unstrukturiertes UI
- ◆ Unübersichtliches UI
- ◆ Unsinniges UI
- ◆ Unklares UI

UI kurz und knapp



- ◆ **UI-Design ist kein ZEN und kein Tao**
- ◆ **Dieser Vortrag enthält keine „Wege zum „UI-Himmel“**
- ◆ **Es gibt auch keine „Bullets für UI-Design“, die es abzuhaken gilt**
- ◆ **Er enthält nur ein paar grundlegende Regeln und Tipps**
- ◆ **Ferner sind ein paar Do's und Dont's enthalten**

Erste Fragen

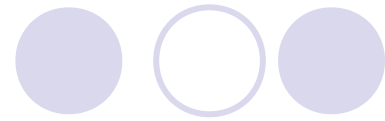


- ◆ **Für wen ist das UI ?**
- ◆ **Was soll über das UI zur Verfügung stehen ?**
- ◆ **Wie kann dem Anwender eine optische Hilfestellung gegeben werden ?**
- ◆ **Ist das UI vollständig UND übersichtlich ?**



- ◆ **Der Anwender erschließt und nutzt die Anwendungsfunktionalität über das UI**
- ◆ **Das UI soll dem Anwender Hilfestellung bei der Nutzung der Anwendung bieten**
- ◆ **Ein gutes UI hilft dem Anwender und verwirrt ihn nicht**
- ◆ **Viele Programmierer begehen „UI-Mord“ oder „Featureitis“**

UI-Regel 1: Erwartung erfüllen



**Ein User Interface ist dann gut
designed, wenn es genau das tut,
was der Anwender erwartet**

◆ **Bsp: Mac vs. PC UI**

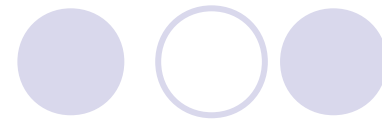
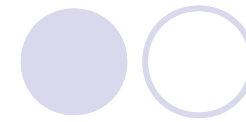
UI-Regel 2: Modellkongruenz



**Ein User Interface ist dann gut
designed, wenn sich das Programm
genau so verhält wie der Anwender
es erwartet**

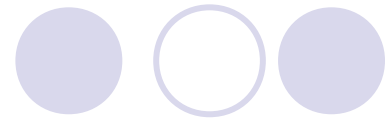
Oder

**Ein User Interface ist dann gut
designed, wenn das Programm-
Modell dem User-Modell entspricht**



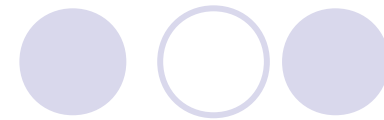
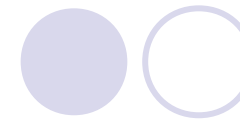
- ◆ **Gutes UI-Design bedeutet, Programm-Modell und User-Modell in Einklang zu bringen**
- ◆ **Wie kommt man an das User-Modell: 5-6 „Testuser“ reichen !!**
- ◆ **Anwender gehen IMMER vom einfachsten anzunehmenden Modell aus**

UI-Regel 3: Sinnvolle Auswahl



- ◆ **Jede Option erfordert vom Anwender eine Auswahl/Entscheidung**
- ◆ **Achten Sie darauf, dass der Anwender nur relevante Entscheidungen trifft und nicht Schlichter des Streites zwischen Designern und Programmierern wird**
- ◆ **Machen Sie nur Sinnvolles zu Optionen oder Änderbar !!**

Regel 3: Beispiel 1



Regel 3: Beispiel 2



mIRC Options [X]

Category:

- ☒ **Connect**
 - Options
 - Local Info
 - Identd
 - Firewall
- ☒ IRC
- ☒ Sounds
- ☒ Mouse
- ☒ DCC
- ☒ Display
 - Options
 - Windows
 - Tray
- ☒ General

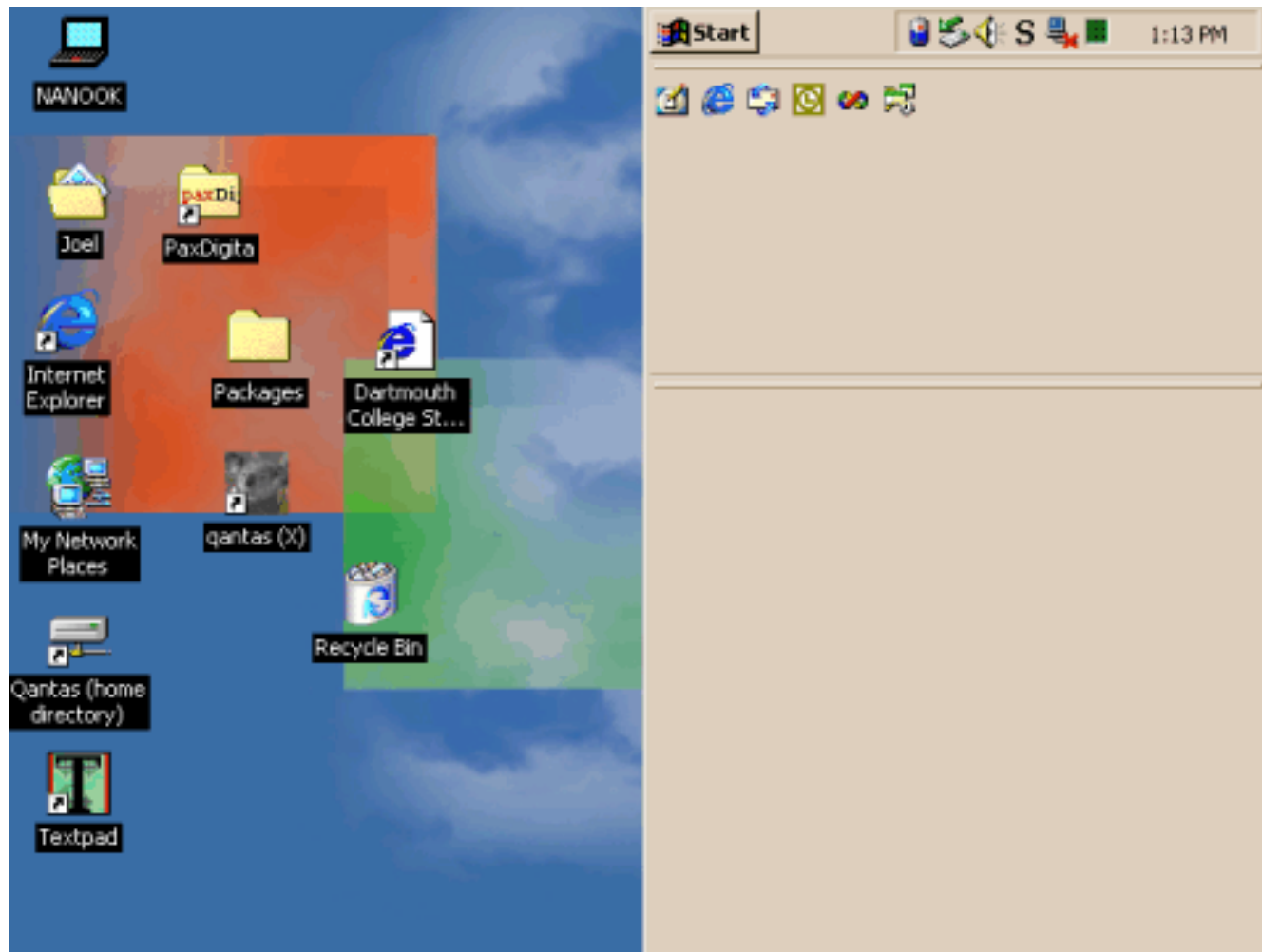
IRC Servers: All [v] [Add]
IRCnet: EU, DE, Berlin [v] [Edit]
[Delete]
[Sort]

[Connect to IRC Server]

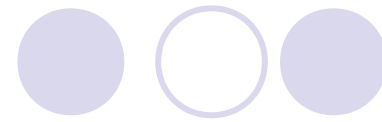
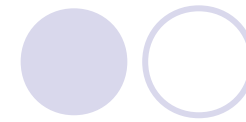
Full Name: Senaj
E-Mail Address: SenajL@DataAssist.de
Nickname: Senaj
Alternative: Senaj@
☐ Invisible mode

[OK] [Cancel] [Help]

Regel 3: Beispiel 4



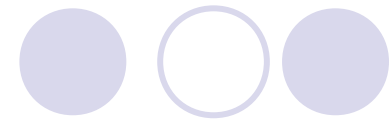
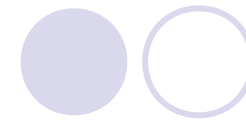
UI-Regel 4: Metaphern



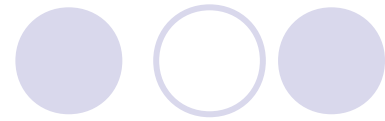
- ◆ **Fehlt das User-Modell oder ist es unvollständig, so zeigen Metaphern dem Anwender das Programm-Modell**
- ◆ **Metaphern sollen sinnvolle optische „Hinweise“ sein**

Eine Metapher weist den Anwender auf rein realitätskonformes Verhalten hin

Regel 4: Beispiel 1



UI-Regel 5: Falsche Metaphern



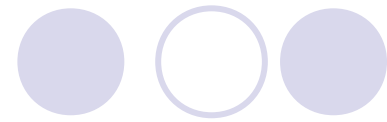
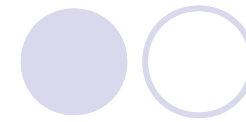
Wenn Metaphern verwendet werden, sollten sie vorhersehbar agieren wie Objekte der realen Welt. Widersprüche zur Realität stiften Verwirrung

- ◆ **Metaphern sollten dem Denkmodell des Anwenders konform gehen**



My Briefcase

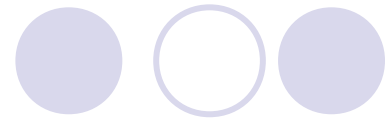
UI-Regel 6: Konsistenz



- ◆ **Die Anwendung sollte sich dem zugrunde Liegenden (Betriebs-) System konform verhalten**

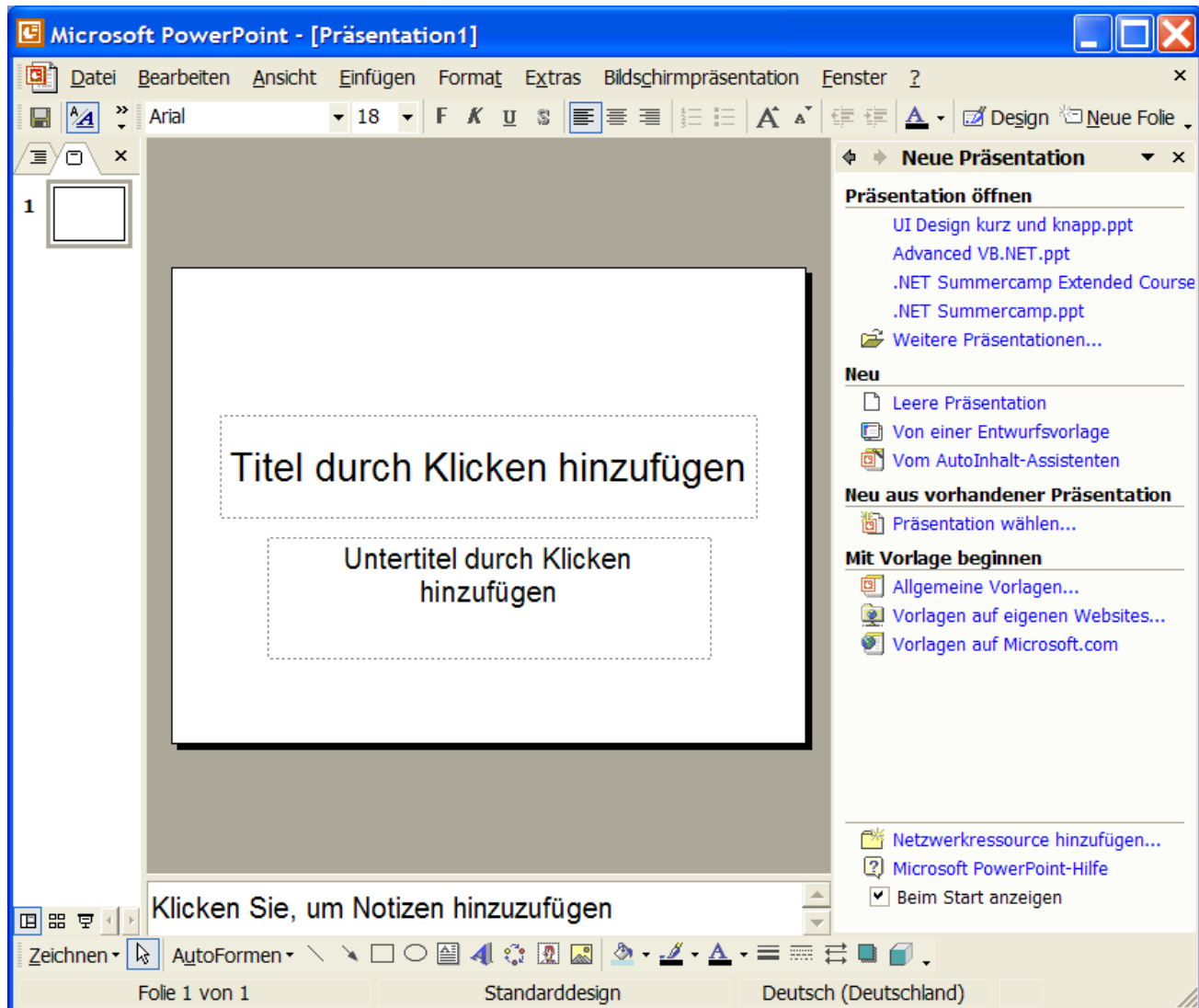
Ein guter UI-Designer wird von Konsistenz gebrauch machen. Das zeigt vielleicht nicht seine Kreativität aber die Anwender werden es lieben

Regel 7: Den Anwender führen

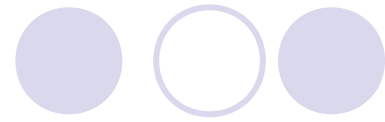


- ◆ Ein gutes UI führt und leitet den Anwender bei seiner Aufgabe und unterstützt ihn dabei
- ◆ Der Anwender KANN nichts falsch machen
- ◆ Die Aufgabe ist unmissverständlich

Regel 7: Beispiel



UI-Regel 8: Design for Extremes



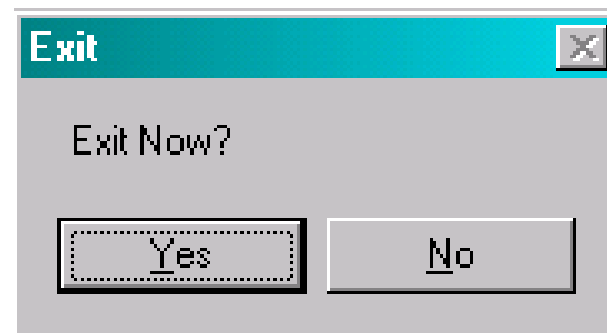
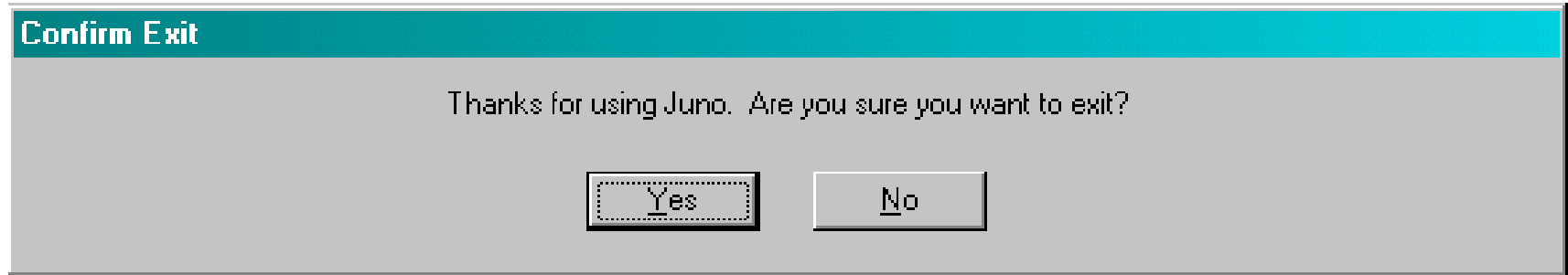
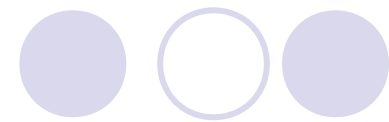
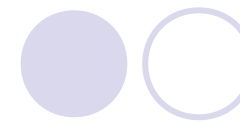
- ◆ Anwender die nicht lesen können
- ◆ Anwender die schlecht lesen können
- ◆ Anwender die mit der Maus nicht so firm sind oder sie nicht bedienen können
- ◆ Anwender die vergesslich sind

8.1: Anwender lesen nicht



- ◆ **Anwender lesen nicht**
- ◆ **Anwender lesen NIE (schon gar nicht das Handbuch)**
- ◆ **Anwender lesen eigentlich nichts (und schon gar keine langen oder umfangreichen Dialoge)**
- ◆ **(Zu viele) Bestätigungsdialoge nerven nur**

8.1: Beispiel 1



8.2: Anwender und die Maus

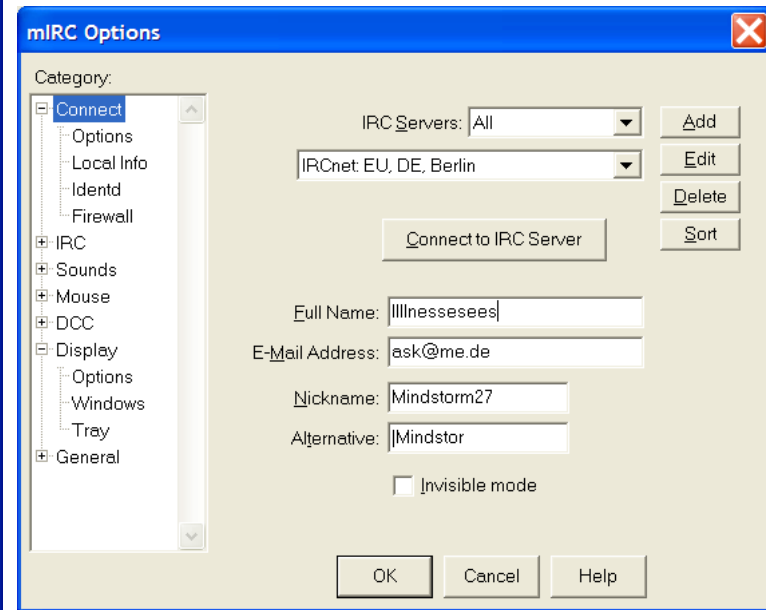
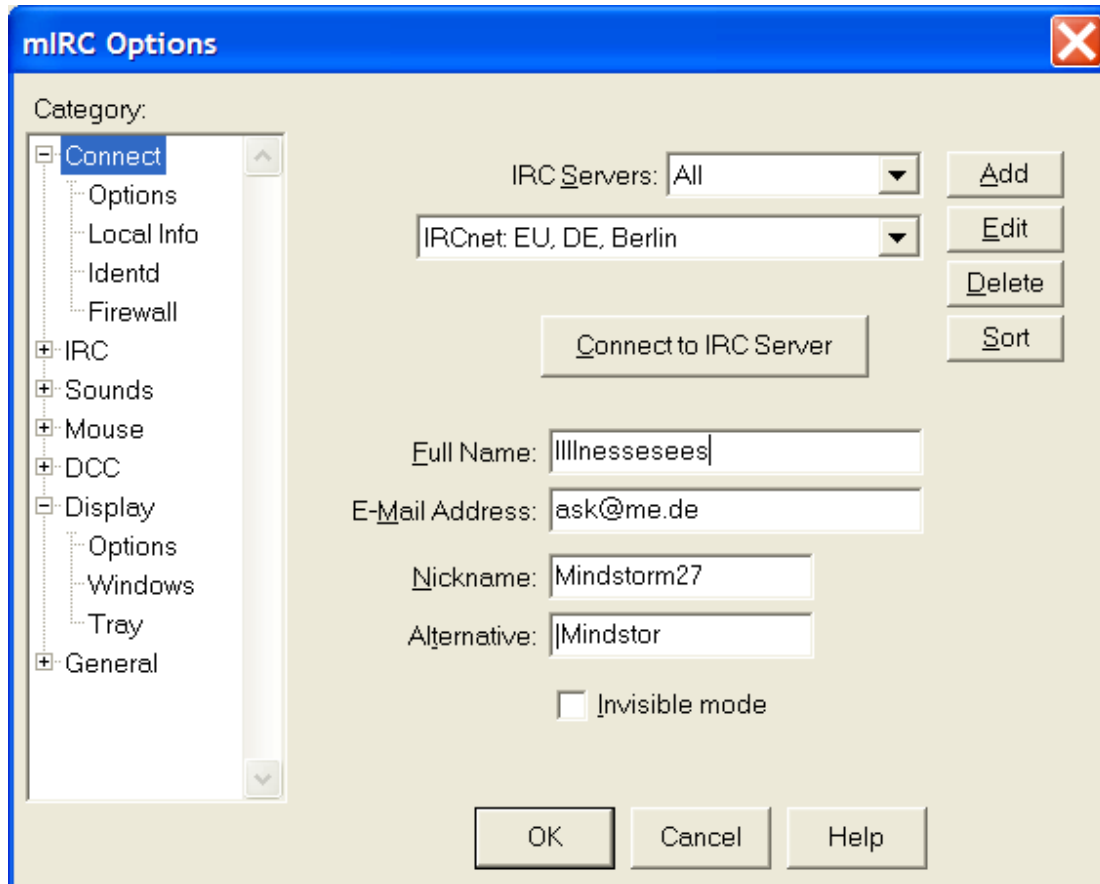
Anwender können die Mäuse nicht bedienen

- ◆ **Viele verschiedene Mäuse (Trackball)**
- ◆ **Mäuse überall (Jeans, Glasoberfläche)**
- ◆ **Einsteiger**
- ◆ **Motorische Hemmnisse**
- ◆ **Doppelklick: Üben oder vermeiden**
- ◆ **Tastaturfanatiker**

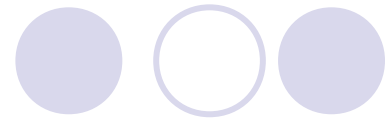
8.3: Anwender und Fonts



- ◆ Bei zu kleiner Schrift wird es unleserlich



8.4: Anwender-Vergesslichkeit

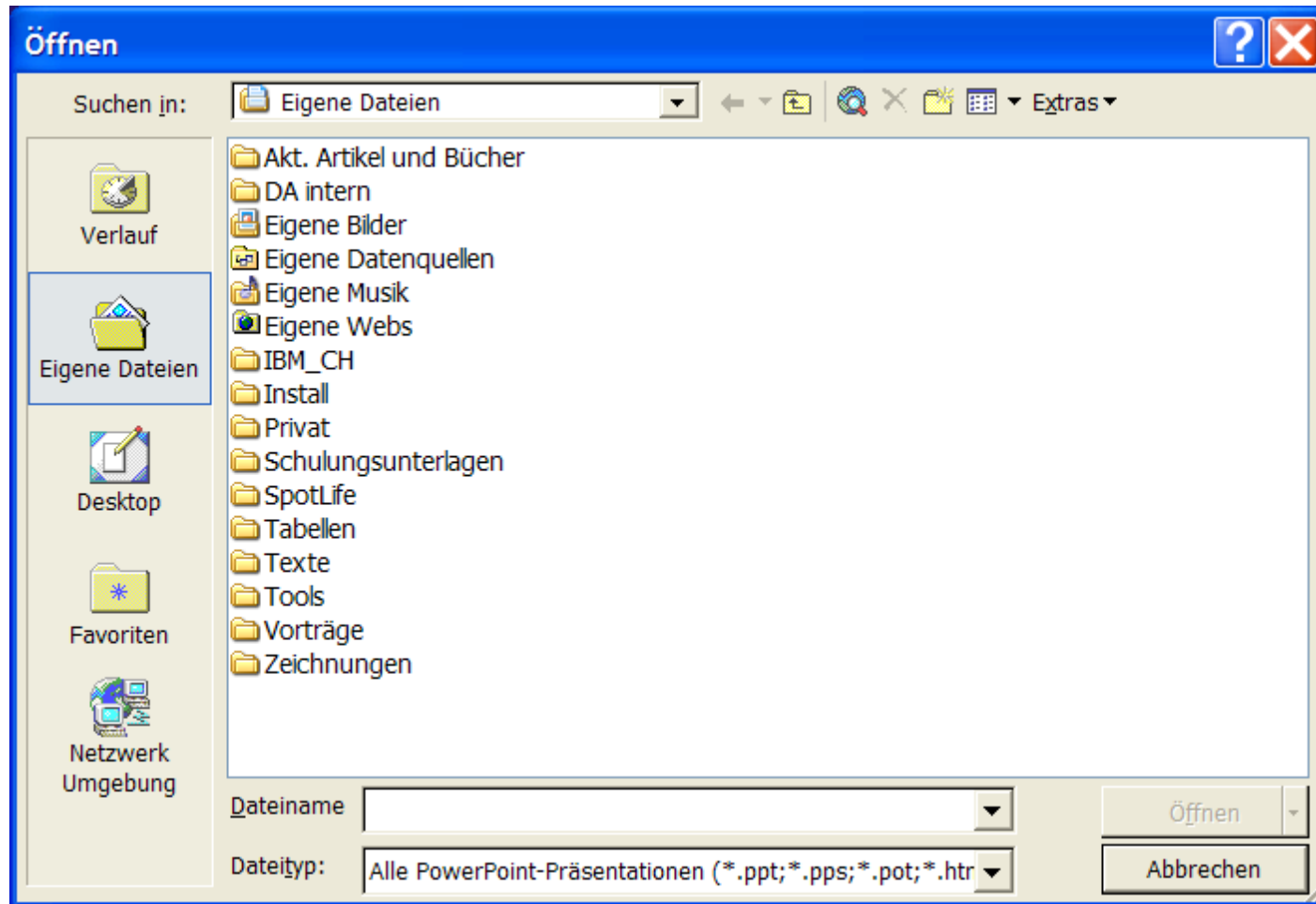


- ◆ **Kommandozeilenoberflächen sind für Anwender „ein Horror“**
- ◆ **Beispiel: DOS-Eingabeaufforderung**

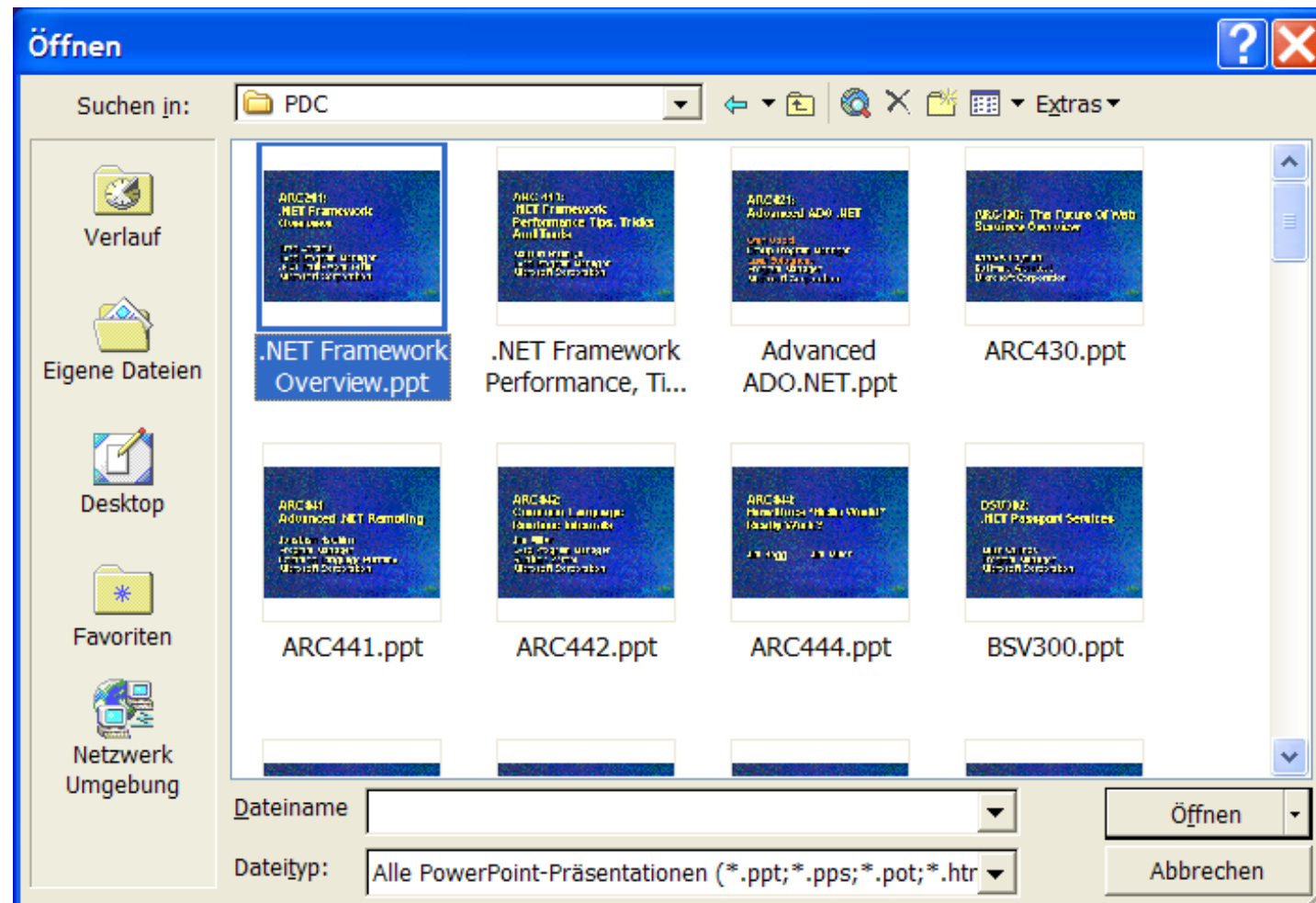
```
Eingabeaufforderung
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Dokumente und Einstellungen\Senaj>
```

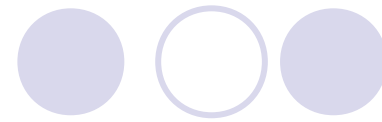
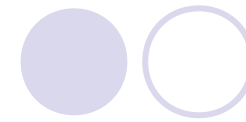
8.4: Beispiel 1



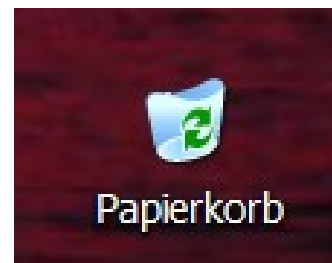
8.4: Beispiel 1 - 2



Hinweise zum UI-Design



- ◆ Anwender-Modell und Profile erstellen
- ◆ Anwender führen (Powerpoint-Dialog)
- ◆ Unerwartete Konsequenzen erwarten (Papierkorb muss „ordentlich“ sein)
- ◆ Test: 5-6 Anwender reichen (statistisch)
- ◆ UI klar und reich an Metaphern machen
- ◆ Nicht spielen – Kreativität ist schlecht



Literatur

User Interface Design for Programmers

Joel Spolsky, Apress, ISBN 1-893115-94-1

Windows User Experience

MS-Press, 0-7356-0566-1

GUI-Design

Ivo Wessel, Hanser, 3-446-19389-8